

# **Fachhochschule Münster University of Applied Sciences**

**Modulhandbuch**

**der beruflichen Fachrichtung Informatik  
(fachwissenschaftlicher Teil)**

**für die Bachelorstudiengänge**

**Berufliche und allgemeine Bildung (BAB)  
und  
Berufliche Bildung (BB)**

**(Stand: 18. Dezember 2007)<sup>1</sup>**

**Fachhochschule Münster, Fachbereich Elektrotechnik und Informatik  
Stegerwaldstraße 39, 48 565 Steinfurt, Tel.: +49 2551 9-62123  
eMail: [eti@fh-muenster.de](mailto:eti@fh-muenster.de), [http: www.fh-muenster.de/fb2](http://www.fh-muenster.de/fb2)**



---

<sup>1</sup>Die aktuellste Version steht Online unter <https://www.fh-muenster.de/fb2/studierende> als pdf-Version zum Download zur Verfügung.

# Lehramt an Berufskollegs

## Berufliche Fachrichtung Informatik

### Ziele

Das Studium vermittelt unter Beachtung der allgemeinen Studienziele auf der Grundlage wissenschaftlicher Erkenntnisse und Methoden sowohl theoretische als auch anwendungsbezogene Inhalte des Studienfachs und befähigt dazu, Vorgänge und Probleme aus den Berufsfeldern der Informatik zu analysieren, praxisgerechte Lösungen zu erarbeiten und dabei auch außerfachliche Bezüge zu beachten.

### Berufsfelder

Die Tätigkeit des Lehrers im höheren Lehramt an beruflichen Schulen verlangt eine hohe Fachkompetenz in den ingenieur- und naturwissenschaftlichen Disziplinen. Darüber hinaus übernimmt der Lehrer aber auch vielfältige pädagogische Bildungs- und Erziehungsaufgaben, die weit über das Fachliche hinausgehen. Demzufolge benötigt ein Lehrer folgende Kompetenzen: Fachwissen, Methodik und Didaktik, Pädagogik und Persönlichkeit.

### Studium

#### Vorlesungen; Übungen; Praktika

Die Vorlesungen dienen zur Vermittlung und gemeinsamen Erarbeitung der Fachlichkeit. Dabei wird von den Dozenten bevorzugt ein seminaristischer Vorlesungsstil eingesetzt. Bis auf Grundlagenveranstaltungen des Grundstudiums werden die meisten Vorlesungen in kleineren Gruppen von nicht mehr als 30 bis 40 Studenten abgehalten.

Übungen ermöglichen den Studenten die Anwendung des neu gewonnenen Wissens und vertiefen mithin das Fachwissen sowie insbesondere die Methodenkompetenz. Im Allgemeinen werden die Lösungen der gestellten Aufgaben nach einer Zeit für die selbständige Lösung gemeinsam erarbeitet.

In den Praktika, die in fortgeschrittenen Lehrveranstaltungen als Projektveranstaltungen mit abschließender Ergebnispräsentation durchgeführt werden können, hat jeder Student einen eigenen Arbeitsplatz, selten müssen sich zwei Studenten einen Arbeitsplatz teilen.

Die Anwendungsbeispiele in Vorlesungen und Übungen sowie insbesondere die Aufgabenstellungen der Praktika stammen in aller Regel aus einem konkreten Anwendungsbereich, so dass bereits hier ein gewisses Maß an Interdisziplinarität vermittelt wurde.

Die Prüfungen erfolgen in allen Studiengängen studienbegleitend am Ende eines Moduls. Neben schriftlichen oder mündlichen Prüfungen gibt es auch besondere Prüfungsformen, wie z.B. Hausarbeiten, Projektarbeiten oder Präsentationen. Zum Abschluss eines Projektpraktikums ist eine professionelle Projektpräsentation als Prüfungsleistung vorgesehen. Sind externe Partner an dem Projekt beteiligt, kann die Präsentation als Prüfungsleistung auch extern erfolgen.

## **Abschluss**

Nach erfolgreicher Beendigung des Bachelorstudiums wird von der Fachhochschule Münster der Hochschulgrad „Bachelor of Science“ Kurzbezeichnung „B.Sc.“ vergeben.



# Studienverlaufspläne

## Abkürzungen

V = Vorlesung  
 Ü = Übung  
 P = Praktikum  
 SWS = Semesterwochenstunden

LP = Leistungspunkte  
 PA = Prüfungsart  
 MP = Modulprüfung  
 TN = Teilnahmenachweis

## Bemerkungen

Eine Modulprüfung setzt immer einen Teilnahmenachweis als Zulassung zur Prüfung voraus.

## Basismodule

Modul	1. Semester					2. Semester					3. Semester						
	V	Ü	P	LP	PA	V	Ü	P	LP	PA	V	Ü	P	LP	PA	SWS	LP
Informatik I	2	3	1	7	MP											6	7
Informatik II						2	3	1	7	MP						6	7
Mathematik I	5	2	0	8	MP											7	8
Mathematik II						4	2	0	7	MP						6	7
Datenbanken						2	0	2	6	MP						4	6
Summe	7	5	1			8	7	1									
<b>Summe der Module</b>	13		18			16		20							29	35	

## Vertiefungsmodule

Modul	3. Semester					4. Semester					5. Semester					SWS LP	
	V	Ü	P	LP	PA	V	Ü	P	LP	PA	V	Ü	P	LP	PA		
Algorithmen und Datenstrukturen	3	1	2	8	MP											6	8
Mikroprozessortechnik	2	1	0	4		1	0	2	5	MP						6	9
Projekt Softwareentwicklung (LPO-konform)	0	0	2	4		0	0	2	4	MP						4	8
Betriebssysteme (LPO-konform)						2	1	2	7		2	0	2	7	MP	9	14
Software Engineering											2	0	3	7	MP	5	7
1. Wahlmodul						4	0	0	7	MP						4	7
2. Wahlmodul											4	0	0	7	MP	4	7
Summe	5	2	4			3	5	6			4	4	5				
<b>Summe der Module</b>	11		16			14		23			13		21			38	60
<b>Summe 1.-5. Sem.</b>																67	95

## Wahlmodule

Modul	Richtung	4. Semester					5. Semester					SWS LP	
		V	Ü	P	LP	PA	V	Ü	P	LP	PA		
Bussysteme		2	0	2	7	MP						4	7
Daten- und Netzwerksicherheit							2	0	2	7	MP	4	7
Kommunikationssysteme		3	1	1	7	MP						5	7
Local Area Networks		2	1	1	7	MP						4	7
Mathematik III für Informatiker							4	2	0	7	MP	6	7
Netzwerkprogrammierung							2	0	2	7	MP	4	7
Objektorientierte Systeme		2	0	2	7	MP						4	7
Projektmanagement						1	2	1	4	MP	4	7	
Rechnerarchitektur		2	0	2	7	MP	2	0	2	7	MP	8	14

# Modulverzeichnis

<b>1 Basismodule</b>	<b>6</b>
1.1 Modul Datenbanken . . . . .	6
1.2 Informatik . . . . .	8
1.2.1 Modul Informatik I . . . . .	8
1.2.2 Modul Informatik II . . . . .	10
1.3 Mathematik . . . . .	12
1.3.1 Modul Mathematik I . . . . .	12
1.3.2 Modul Mathematik II . . . . .	14
<b>2 Vertiefungssmodule</b>	<b>16</b>
2.1 Modul Algorithmen und Datenstrukturen . . . . .	16
2.2 Modul Betriebssysteme . . . . .	18
2.3 Modul Mikroprozessortechnik . . . . .	20
2.4 Projekt Softwareentwicklung . . . . .	22
2.4.1 Modul Projekt Softwareentwicklung I . . . . .	22
2.4.2 Modul Projekt Softwareentwicklung II . . . . .	24
2.5 Modul Software Engineering . . . . .	26
<b>3 Wahlmodule</b>	<b>28</b>
3.1 Modul Bussysteme . . . . .	28
3.2 Modul Daten- und Netzwerksicherheit . . . . .	30
3.3 Modul Kommunikationssysteme . . . . .	32
3.4 Modul Local Area Networks . . . . .	34
3.5 Mathematik . . . . .	36
3.5.1 Modul Mathematik III für Informatiker . . . . .	36
3.6 Modul Netzwerkprogrammierung . . . . .	38
3.7 Modul Objektorientierte Systeme . . . . .	40
3.8 Modul Projektmanagement . . . . .	42
3.9 Modul Rechnerarchitektur . . . . .	44

# 1 Basismodule

## 1.1 Modul Datenbanken

Kennnummer: L IT 1.1		Aufwand: LIT 180 h	Leistungs- punkte: LIT 6 LP	Studiense- mester: LIT 2.	Dauer: 1 Semester
1.	Lehrveranstaltung(en): Datenbanken	Kontaktzeit: 2+0+2 SWS		Selbststudium: LIT 116 h	LP:
2.	Lehrformen:	Vorlesung+Übung+Praktikum: 2+0+2 SWS			
3.	Gruppengröße:	Vorlesung: ca. 36, Übung: 0, Praktikum: ca. 3 x 12			
4.	Qualifikationsziele:	Die Studierenden kennen die grundlegenden Konzepte von Datenbanksystemen. Sie sind damit in der Lage, Datenbanken zu entwickeln und zu realisieren.			
5.	Inhalte:	<p>Grundlegende Konzepte</p> <p>Architekturen von Datenbanksystemen</p> <p>SQL kompakt</p> <p>Modelle zum Datenbankentwurf: ER, UML</p> <p>Modelle für die Realisierung: Relationenmodell, hierarchisches Modell, Netzwerkmodell, erweiterte relationale, semantische, objektorientierte und objekt-relationale Modelle</p> <p>Datenbankentwurf: Relationaler Datenbankentwurf: Abhängigkeiten, Schemaeigenschaften, Transformationseigenschaften etc.</p> <p>Datenbankdefinitionssprachen: SQL-DDL, CODASYL-DDL, IMS-DDL, ODL</p> <p>Grundlagen von Anfragesprachen: Interaktives SQL</p> <p>Datenbank-Anwendungsprogrammierung: Transaktionsmodelle und Transaktionsverwaltung, Dateiorganisation und Zugriffsstrukturen, Wiederherstellung und Datensicherheit</p> <p>Weitere SQL-Features und Datenbankthemen: Sequenzen, Large Objects, Zugriffsrechte, Trigger, OR-Features von SQL-99, Data Dictionaries, Anfrageoptimierung, Data Warehousing, Verteilte Datenbanken und Verteilte Informationssysteme</p>			

		<p>Praktikum: SQL kompakt; ER-Datenmodellierung; Datenmodellierung relational und hierarchisch; Abbildung ER-Modell in relationales, hierarchisches und Netzwerkmodell; Normalformen, funktionale und mehrwertige Abhängigkeiten; DB2-SQL I, DB2-SQL II; DB2-SQL III; Stored Functions und Procedures; SQLJ</p> <p>Fachliteratur (Auswahl): [1] Heuer, Saake: Datenbanken: Konzepte und Sprachen. 2. Auflage. International Thomson Publishing 2000. [2] Saake, Heuer: Datenbanken: Implementierungstechniken. International Thomson Publishing 1999. [3] Heuer, Saake, Sattler: Datenbanken kompakt. International Thomson Publishing 2001. [4] Vossen: Datenbankmodelle, Datenbanksprachen und Datenbankmanagementsysteme. Oldenbourg 2000</p>
6.	Teilnahmevoraussetzungen:	Inhaltlich baut das Modul auf die Veranstaltungen Informatik I und Informatik II sowie Algorithmen und Datenstrukturen auf. Elementare Rechnerstrukturen, Programmierkenntnisse in Java und C, Algorithmen und Datenstrukturen, Grundlagen der Komplexitätstheorie sind notwendig.
7.	Prüfungsformen:	Klausur oder mündliche Prüfung
8.	Voraussetzungen für die Vergabe von Kreditpunkten:	- An- und Abtestate der Praktikumsaufgaben - Bestehen der Prüfung
9.	Häufigkeit des Angebots:	Jährlich
10.	Modulbeauftragter: Hauptamtlich Lehrende: Lehrbeauftragte:	Prof. Dr. Thomas Weik Prof. Dr. Thomas Weik —
11.	Sonstige Informationen:	

## 1.2 Informatik

### 1.2.1 Modul Informatik I

Kennnummer: L IT 1.2.1		Aufwand: LIT 210 h	Leistungs- punkte: LIT 7 LP	Studiense- mester: LIT 1.	Dauer: 1 Semester
1.	Lehrveranstaltung(en): Informatik I	Kontaktzeit: 2+3+1 SWS		Selbststudium: LIT 144 h	LP:
2.	Lehrformen:	Vorlesung+Übung+Praktikum: 2+3+1 SWS			
3.	Gruppengröße:	Vorlesung: ca. 150, Übung: ca. 2 x 75, Praktikum: ca. 7 x 20			
4.	Qualifikationsziele:	Die Studierenden kennen die Aufteilung der Informatik in ihre Teilgebiete und lernen die grundlegenden Denkweisen, Verfahren und Grenzen der Informatik. Sie lernen, Abläufe durch Algorithmen zu spezifizieren und zu implementieren und verstehen, welche Rolle Abstraktion und Modellbildung in der Informatik spielen. Die Entwicklung von Software mithilfe der Programmiersprache C wird ermöglicht, der praktische Umgang mit Rechnern wird eingeübt.			
5.	Inhalte:	<p>Grundlagen: Begriff Informatik; Informationen und Daten; Informationsdarstellung; Kodierung, Ganzzahldarstellungen; Gleitkommazahlen, Soft-, Firm- und Hardware.</p> <p>Rechnerarchitekturen: Boolesche Algebra, Maschinensprache; Betriebssysteme, Bootprozess; Anwendungsprogramme, Multitasking; elementare Datensicherheit (Verschlüsselung, Authentisierung)</p> <p>Programmierung: Programmiersprachen, Interpreter, Compiler und Assembler; Spezifikationen, Algorithmen und Programme; Daten und Datenstrukturen; imperative Sprachen; Formale Beschreibung von Programmiersprachen; Unterprogramme; Rekursion; Konstruktion neuer Datentypen; strukturierte Programmierung, Kurzeinführung in objektorientierte Programmierung</p> <p>Algorithmen und Datenstrukturen: Elementare Datenstrukturen; Arrays, Bäume; Stacks, Queues, Listen, Graphen, Analyse von Algorithmen; Sortieralgorithmen; Suchalgorithmen, graphentheoretische Algorithmen (z. B. Dijkstra)</p> <p>Theoretische Informatik: Einführung und Historie; Endliche Automaten; Turing-Maschinen; Unentscheidbare Probleme; Grammatiken; Komplexität; Klassen P und NP.</p> <p>Praktikum: Programmierkurs C</p>			

		Fachliteratur (Auswahl): 1] H.-P. Gumm, M. Sommer: Einführung in die Informatik, Oldenbourg, 5. Auflage, 2006 [2] R. Sedgewick: Algorithmen in C, Addison-Wesley, 1. Auflage, 1998. [3] B. W. Kernighan, D. M. Ritchie: The C Programming Language, Prentice Hall, Second Edition, 1988 [4] J. E. Hopcroft, R. Motwani, J. D. Ullman: Einführung in die Automatentheorie, Formale Sprachen und Komplexitätstheorie, Addison-Wesley, 2. Auflage, 2002.
6.	Teilnahmevoraussetzungen:	Keine
7.	Prüfungsformen:	Klausur oder mündliche Prüfung
8.	Voraussetzungen für die Vergabe von Kreditpunkten:	- An- und Abtestate der Praktikumsaufgaben - Bestehen der Prüfung
9.	Häufigkeit des Angebots:	Jährlich
10.	Modulbeauftragter: Hauptamtlich Lehrende:  Lehrbeauftragte:	Prof. Dr. Nikolaus Wulff Prof. Dr. Nikolaus Wulff Prof. Dr. Thomas Weik Prof. Dr. Ulrich Greveler —
11.	Sonstige Informationen:	

**1.2.2 Modul Informatik II**

Kennnummer: L IT 1.2.2		Aufwand: LIT 210 h	Leistungs- punkte: LIT 7 LP	Studiense- mester: LIT 2.	Dauer: 1 Semester
1.	Lehrveranstaltung(en): Informatik II	Kontaktzeit: 2+3+1 SWS		Selbststudium: LIT 144 h	LP:
2.	Lehrformen:	Vorlesung+Übung+Praktikum: 2+3+1 SWS			
3.	Gruppengröße:	Vorlesung: ca. 150, Übung: ca. 2 x 75, Praktikum: ca. 7 x 20			
4.	Qualifikationsziele:	Aufbauend auf Informatik I kennen die Studierenden die Aufteilung der Informatik in ihre Teilgebiete und lernen die grundlegenden Denkweisen und Verfahren der Informatik sowie die Programmiersprache Java kennen. Sie sind in der Lage, Softwareentwicklungsprojekte in Java durchzuführen.			
5.	Inhalte:	<p>Rechnerarchitektur II: Von den Schaltgliedern zur CPU; Assemblerprogrammierung.</p> <p>Betriebssysteme: Einführung; Verwaltung von Ressourcen; UNIX; UNIX-Prozesse; X Window-System; MS Windows; Andere Betriebssysteme.</p> <p>Rechnernetze: Rechner-Verbindungen; Datenübertragung über Telefonleitungen; Protokolle und Netze; Netztechnologien.</p> <p>Internet und XML: TCP/IP; Dienste im Internet; Das WWW; Einführung in XML.</p> <p>Compilerbau: Programmiersprachen; Programmierung eines Compilers: Lexikalische Analyse, Syntaxanalyse und Codegenerierung.</p> <p>Datenbanksysteme: Anforderungen; Architektur; Datenmodelle; SQL</p> <p>Graphikprogrammierung: Hardware; Graphikroutinen für Rastergraphik; Programmierbeispiele; 3-D-Graphikprogrammierung.</p> <p>Software Engineering: Werkzeuge und Methoden für Software-Projekte; Vorgehensmodelle; OOSE; Der Rational Unified Process; Projekt-Management; Qualitätsmanagement; Aktuelle Werkzeuge und Umgebungen</p> <p>Praktikum: Programmierkurs Java</p>			

		Fachliteratur (Auswahl): [1] H.-P. Gumm, M. Sommer: Einführung in die Informatik, Oldenbourg, 6. Auflage, 2004. [2] Barnhes, Kölling:Objektorientierte Programmierung mit Java, Pearson SStudio, 1. Auflage 2003. [3] H. Balzert: Lehrbuch der Software-Technik: Software-Entwicklung, Spektrum, 2. Auflage, 2000. [4] A. Heuer, G. Saake, K.-U. Sattler: Datenbanken kompakt, International Thomson Publishing, 1. Auflage 2001.
6.	Teilnahmevoraussetzungen:	Informatik I
7.	Prüfungsformen:	Klausur oder mündliche Prüfung
8.	Voraussetzungen für die Vergabe von Kreditpunkten:	- An- und Abtestate der Praktikumsaufgaben - Bestehen der Prüfung
9.	Häufigkeit des Angebots:	Jährlich
10.	Modulbeauftragter: Hauptamtlich Lehrende:  Lehrbeauftragte:	Prof. Dr. Nikolaus Wulff Prof. Dr. Nikolaus Wulff Prof. Dr. Thomas Weik Prof. Dr. Ulrich Greveler —
11.	Sonstige Informationen:	

## 1.3 Mathematik

### 1.3.1 Modul Mathematik I

Kennnummer:		Aufwand:	Leistungs- punkte:	Studiense- mester:	Dauer:
L IT 1.3.1		240 h	8 LP	1.	1 Semester
1.	Lehrveranstaltung(en): Mathematik I		Kontaktzeit: 5+2+0 SWS	Selbststudium: 128 h	LP:
2.	Lehrformen:	Vorlesung+Übung+Praktikum: 5+2+0 SWS			
3.	Gruppengröße:	Vorlesung: ca. 150, Übung: ca. 2 x 75, Praktikum: 0			
4.	Qualifikationsziele:	Die Studierenden beherrschen die grundlegenden Begriffe und Methoden der höheren Mathematik in den Teilgebieten Differentialrechnung und Lineare Algebra. Sie verfügen über die Kompetenz zur Anwendung der mathematischen Begriffe und Methoden in weiterführenden Veranstaltungen der Elektrotechnik und Informatik.			
5.	Inhalte:	<p>Logik und Mengenlehre: Aussagenlogik, Prädikatenlogik, mathematische Beweise, Mengenbegriff, Mengenoperationen, Relationen und Abbildungen</p> <p>Zahlen: natürliche und ganze Zahlen, natürliche Zahlen und vollständige Induktion, Binomialkoeffizienten, ganze, rationale und reelle Zahlen, komplexe Zahlen</p> <p>Folgen und Reihen: Folgen, Grenzwerte, Eulersche Zahl, Rechnen mit Grenzwerten, Reihen</p> <p>Funktionen einer reellen Variablen: Definition und Darstellung, einfache Funktionen, Umkehrfunktion und Verkettung, Funktionsklassen, ganze und gebrochenrationale Funktionen, Potenz- und Wurzelfunktionen, Exponential- und Logarithmusfunktion, Trigonometrische und Arcusfunktionen, Hyperbel- und Areafunktionen, Grenzwerte, Stetigkeit, Eigenschaften stetiger Funktionen</p> <p>Einführung in komplexe Funktionen: trigonometrische Darstellung komplexer Zahlen, komplexe Exponentialfunktion, komplexe Funktionen</p> <p>Matrizen, Determinanten und Vektoralgebra: Matrizen, Addition und Multiplikation, inverse Matrix, Determinanten, Vektoralgebra, Lineare Abhängigkeit, Basis, Skalar- und Vektorprodukt, Anwendungen Lineare Gleichungssysteme und Eigenwerte: Gaußscher Algorithmus, Eigenwerte und Eigenvektoren</p>			

		<p>Differentialrechnung:  Definition der Ableitung, Ableitungsregeln, Linearkombination, Produkt- und Quotientenregel, Kettenregel, Ableitung der Umkehrfunktion, Höhere Ableitungen, Ableitung elementarer Funktionen, rationale Funktionen, trigonometrische Funktionen, Exponential- und Logarithmusfunktion, hyperbolische Funktionen, Mittelwertsatz der Differentialrechnung, Regel von de l'Hospital, Kurvendiskussion, Konvergenz von Reihen Potenzreihen, Satz von Taylor</p> <p>Fachliteratur (Auswahl):  [1] A. Fetzner, H. Fränkel, Mathematik, Lehrbuch für ingenieurwissenschaftliche Studiengänge, 2 Bände, Springer, Berlin, 2004  [2] Lothar Papula, Mathematik für Ingenieure und Naturwissenschaftler, 3 Bände, Vieweg Verlag, Braunschweig, 2001  [3] T. Westermann, Mathematik für Ingenieure mit Maple, 2 Bände, Springer, Berlin, 2004  [4] H. Anton, Calculus, John Wiley &amp; Sons, New York, 2002</p>
6.	Teilnahmevoraussetzungen:	Kenntnisse aus der Differential- und Integralrechnung, Linearen Algebra und Wahrscheinlichkeitsrechnung auf dem Niveau eines Grundkurses der Sekundarstufe II
7.	Prüfungsformen:	Klausur
8.	Voraussetzungen für die Vergabe von Kreditpunkten:	Bestehen der Prüfung
9.	Häufigkeit des Angebots:	Jährlich
10.	Modulbeauftragter: Hauptamtlich Lehrende: Lehrbeauftragte:	Prof. Dr. Gernot Bauer Prof. Dr. Gernot Bauer Prof. Dr. Hans Effinger —
11.	Sonstige Informationen:	

## 1.3.2 Modul Mathematik II

Kennnummer:		Aufwand:	Leistungs- punkte:	Studiense- mester:	Dauer:
L IT 1.3.2		210 h	7 LP	2.	1 Semester
1.	Lehrveranstaltung(en): Mathematik II		Kontaktzeit: 4+2+0 SWS	Selbststudium: 98 h	LP:
2.	Lehrformen:	Vorlesung+Übung+Praktikum: 4+2+0 SWS			
3.	Gruppengröße:	Vorlesung: ca. 150, Übung: ca. 2 x 75, Praktikum: 0			
4.	Qualifikationsziele:	Die Studierenden beherrschen die grundlegenden Begriffe und Methoden der höheren Mathematik in den Teilgebieten Integralrechnung, Funktionen mehrerer Variablen und Differentialgleichungen. Sie verfügen über die Kompetenz zur Anwendung der mathematischen Begriffe und Methoden in weiterführenden Veranstaltungen der Elektrotechnik und Informatik.			
5.	Inhalte:	<p>Integralrechnung: Definition und Eigenschaften des bestimmten Integrales, Stammfunktionen, Hauptsatz der Differential- und Integralrechnung, Grundintegrale, Substitutionsmethode, partielle Integration, Partialbruchzerlegung und Integration gebrochenrationaler Funktionen, numerische Integration, uneigentliche Integrale, Integration von Potenzreihen, Inhalt ebener Flächen, Volumen und Mantelfläche von Rotationskörpern, Bogenlänge ebener Kurven, Mittelwerte, Schwerpunkt homogener Flächen und Körper</p> <p>Funktionen mehrerer Variablen: Definition und Darstellungsformen, Stetigkeit, partielle Ableitung, totale Differenzierbarkeit, Tangentialebenen, totales Differential, Linearisierung von Funktionen, lineare Fehlerfortpflanzung, Kettenregel, Gradient, Richtungsableitung, Extremwerte, Vektorfelder, Wege und Kurven im Raum, Tangentialvektor, Kurvenintegrale, Integrabilitätsbedingungen, Potential</p> <p>Gewöhnliche Differentialgleichungen: Grundbegriffe, Existenz- und Eindeutigkeitssatz, Trennung der Variablen bei separablen Differentialgleichungen 1. Ordnung, Variation der Konstanten bei linearen Differentialgleichungen 1. Ordnung, lineare Differentialgleichungen 2. Ordnung mit konstanten Koeffizienten, charakteristisches Polynom, allgemeine Lösung der homogenen Differentialgleichung, partikuläre Lösung der inhomogenen Differentialgleichung</p> <p>Fachliteratur (Auswahl): [1] A. Fetzer, H. Fränkel, Mathematik, Lehrbuch für ingenieurwissenschaftliche Studiengänge, 2 Bände, Springer, Berlin, 2004 [2] Lothar Papula, Mathematik für Ingenieure und Naturwissenschaftler, 3 Bände, Vieweg Verlag, Braunschweig, 2001 [3] T. Westermann, Mathematik für Ingenieure mit Maple, 2 Bände, Springer, Berlin, 2004 [4] H. Anton, Calculus, John Wiley &amp; Sons, New York, 2002</p>			
6.	Teilnahmevoraussetzungen:	Kenntnisse aus dem Modul Mathematik I			

7.	Prüfungsformen:	Klausur
8.	Voraussetzungen für die Vergabe von Kreditpunkten:	Bestehen der Prüfung
9.	Häufigkeit des Angebots:	Jährlich
10.	Modulbeauftragter: Hauptamtlich Lehrende: Lehrbeauftragte:	Prof. Dr. Hans Effinger Prof. Dr. Gernot Bauer Prof. Dr. Hans Effinger —
11.	Sonstige Informationen:	

## 2 Vertiefungsmodule

### 2.1 Modul Algorithmen und Datenstrukturen

Kennnummer:	Aufwand:	Leistungs- punkte:	Studiense- mester:	Dauer:
L IT 2.1	240 h	8 LP	3.	1 Semester
1.	Lehrveranstaltung(en): Algorithmen und Datenstrukturen	Kontaktzeit: 3+1+2 SWS	Selbststudium: 144 h	LP:
2.	Lehrformen:	Vorlesung+Übung+Praktikum: 3+1+2 SWS		
3.	Gruppengröße:	Vorlesung: ca. 30, Übung: ca. 30, Praktikum: ca. 2 x 15		
4.	Qualifikationsziele:	Die Studierenden sind in der Lage, algorithmische Strukturen in konkreten Problemstellungen zu erkennen und Lösungen in Form von Algorithmen zu finden, auszuwählen, anzupassen und in der Programmiersprache C zu implementieren.		
5.	Inhalte:	<p>Einführung:            Grundlagen und Kurzwiederholung C            Elementare Datenstrukturen            Bäume            Rekursion            Analyse und Implementierung von Algorithmen            Sortieralgorithmen            Suchalgorithmen            Suchen in Zeichenketten            Pattern Matching und Syntaxanalyse            Komprimierung und Kryptologie            Geometrische Algorithmen</p> <p>Algorithmen für Graphen:            Elementare Algorithmen, Zusammenhang, Gerichtete Graphen,            Gewichtete Graphen.            Zufallszahlen            Arithmetik            Gaußsches Eliminationsverfahren            Unentscheidbarkeit und die Klassen P und NP</p> <p>Praktikum:            Queues; Türme von Hanoi; Vergleich von Sortierverfahren; Ha-            shing; Zeichenfolgen; Einfacher geschlossener Pfad; Topologisches            Sortieren.</p>		

		<p>Fachliteratur (Auswahl):</p> <p>[1] R. Sedgewick: Algorithmen in C, Addison-Wesley, 1. Auflage, 1998.</p> <p>[2] B. W. Kernighan, D. M. Ritchie: The C Programming Language, Prentice Hall, Second Edition, 1988</p> <p>[3] T. Ottmann, P. Widmayer: Algorithmen und Datenstrukturen, Spektrum, 4. Auflage, 2002</p> <p>[4] N. Wirth: Algorithmen und Datenstrukturen, Teubner, 3. Auflage, 1983</p> <p>[5] J. E. Hopcroft, R. Motwani, J. D. Ullman: Einführung in die Automatentheorie, Formale Sprachen und Komplexitätstheorie, Addison-Wesley, 2. Auflage, 2002.</p>
6.	Teilnahmevoraussetzungen:	Inhaltlich baut das Modul auf die Veranstaltungen Informatik I und Informatik II auf. Die Teilnehmer sollten die Grundlagen der Programmierung kennen und die Programmiersprache C beherrschen.
7.	Prüfungsformen:	Klausur oder mündliche Prüfung
8.	Voraussetzungen für die Vergabe von Kreditpunkten:	- An- und Abtestate der Praktikumsaufgaben - Bestehen der Prüfung
9.	Häufigkeit des Angebots:	Jährlich
10.	Modulbeauftragter: Hauptamtlich Lehrende: Lehrbeauftragte:	Prof. Dr. Thomas Weik Prof. Dr. Thomas Weik —
11.	Sonstige Informationen:	

## 2.2 Modul Betriebssysteme

Kennnummer: L IT 2.2		Aufwand: 420 h	Leistungspunkte: 14 LP	Studiensemester: 4.+ 5.	Dauer: 2 Semester
1.	Lehrveranstaltung(en): Betriebssysteme	Kontaktzeit: 2+1+2 SWS 2+0+2 SWS		Selbststudium: 138 h 138 h	LP: 7 LP 7 LP
2.	Lehrformen:	Vorlesung+Übung+Praktikum: 4+1+4 SWS			
3.	Gruppengröße:	Vorlesung: ca. 36, Übung: ca. 36, Praktikum: ca. 3 x 12			
4.	Qualifikationsziele:	Die Studierenden kennen den Aufbau, die Mechanismen und die Schnittstellen moderner Betriebssysteme. Sie sind dadurch in der Lage, Entscheidungen über den Einsatz von Betriebssystemen in konkreten Anwendungssituationen zu treffen, Systemschnittstellen bei der Software-Entwicklung gezielt einzusetzen und Komponenten von Betriebssystemen eigenständig zu entwickeln.			
5.	Inhalte:	<p>Einführung: Historische Entwicklung, Betriebssystemkomponenten und Strukturen, vom monolithischen zum Microkernel, Betriebssystemschnittstellen, Hardwarestrukturen, Prozessormodi, privilegierter Modus, Benutzermodus, Interrupts, Speicherverwaltung und Speicherschutz</p> <p>Prozesse und Threads: Prozesskonzept, Zustände, Kontextwechsel, Prozesskommunikation, Konzepte und Realisierung, Nachrichten, Pipes, Shared Memory, Prozesssynchronisation, kritische Abschnitte, TSL-Befehl, Spinlocks, Semaphore, Klassische IPC-Probleme, Threads, User und Kernel Threads</p> <p>Prozess-Scheduling: Definitionen und Konzepte, Prozessorauslastung, preemptives und nonpreemptives Scheduling, Algorithmen, FCFS, SJF, Prioritäten, Round-Robin, Multilevel, Echtzeitbetrieb, unterbrechbare Kernel, Bewertungsmethoden, Fallstudien</p> <p>Signale: Grundlagen, asynchrone Ereignissen, Konzept, klassisches UNIX-Signale, Signalmasken, POSIX-Schnittstelle</p> <p>Dateisystem: Organisation, Dateien, Verzeichnisse, Allokationsstrategien, Free List, FAT, UNIX-Dateisystem, Verwaltung offener Dateien, Dateikennzahlen, Dateitabelle, Inode-Tabelle, NTFS, Master File Table, Journaling, Virtuelle und Netzwerkdateisysteme</p>			

		<p>Speicherverwaltung: Adressbindung, logische und physikalische Adressen, MMU, Speicherschutz und Relokation, dynamische Speicherzuweisung, Prozessauslagerung, Seitenverwaltung, Seitentabellen, TLB, mehrstufiges Paging, Hashing, invertierte Seitentabellen, Segmentierung, Virtueller Speicher, Demand Paging, Seitenfehler, Seitenersetzung, FIFO, LRU, Clock, Working Set</p> <p>Verklemmungen-Deadlocks: Allgemeine Charakterisierung, Betriebsmittelgraph, Vorbeugung, Erkennung, Verhinderung, sicherer Zustand, Bankier Algorithmus</p> <p>Ein- und Ausgabe: I/O-Systeme, Organisation, Struktur und Ablauf, Scheduling bei Plattenzugriffen</p> <p>Betriebssysteme auf Multiprozessorsystemen: Master-Slave Kernel, Spin-Locks, Semaphore, SMP, Programmiermodelle</p> <p>The New Frontiers: Verteilte Betriebssysteme, Fallstudien, Ausblick, Trends</p> <p>Praktikum: Prozessmanagement, Interprozesskommunikation auf Basis von Nachrichtensystemen, Pipes, Prozesssynchronisation mit Semaphoren, Implementation einer Shell, Prozess- und Threadsynchronisation, Leser-Schreiber Kooperation, Rekursives Durchsuchen eines Verzeichnisbaumes, Sperren von Dateiabschnitten, Realisierung von Ein-/Ausgaberoutinen mit Timeout. Entwicklung von Kernelmodulen und eigenen Systemaufrufen unter Linux</p> <p>Fachliteratur (Auswahl): [1] A. Silberschatz, P.B. Galvin, G.Gagne, Operating System Concepts, Addison-Wesley, 7th Edition, 2004 [2] A.S. Tanenbaum, Modern Operating Systems, Prentice Hall, Second Edition, 2001 [3] W. Stallings, Operating Systems: Internals and Design Principles, Prentice Hall, 5th Edition, 2005</p>
6.	Teilnahmevoraussetzungen:	Inhaltlich baut das Modul auf die Veranstaltungen Informatik I und Informatik II auf Elementare Rechnerstrukturen, gute Programmierkenntnisse in C und elementare UNIX-Kenntnisse sind notwendig.
7.	Prüfungsformen:	Klausur oder mündliche Prüfung
8.	Voraussetzungen für die Vergabe von Kreditpunkten:	- An- und Abtestate der Praktikumsaufgaben - Bestehen der Prüfung
9.	Häufigkeit des Angebots:	Jährlich
10.	Modulbeauftragter: Hauptamtlich Lehrende: Lehrbeauftragte:	Prof. Dr. Hans Effinger Prof. Dr. Hans Effinger —
11.	Sonstige Informationen:	Unterlagen zur Vorlesung werden elektronisch verteilt.

## 2.3 Modul Mikroprozessortechnik

Kennnummer: L IT 2.3		Aufwand: 270 h	Leistungspunkte: 9 LP	Studiensemester: 3.+4.	Dauer: 2 Semester
1.	Lehrveranstaltung(en): Mikroprozessortechnik	Kontaktzeit: 2+1+0 SWS 1+0+2 SWS		Selbststudium: 72 h 102 h	LP: 4 LP 5 LP
2.	Lehrformen:	Vorlesung+Übung+Praktikum: 3+1+2 SWS			
3.	Gruppengröße:	Vorlesung: ca. 140, Übung: ca. 140, Praktikum: ca. 8 x 16			
4.	Qualifikationsziele:	Die Studierenden beherrschen die Grundlagen der modernen Mikroprozessortechnik einschließlich der Befehlssatzarchitektur und der Mikroarchitektur typischer Mikroprozessoren. Die Studierenden erlangen das notwendige Rüstzeug zum Entwurf sowie zur Implementierung und Programmierung von Mikroprozessorsystemen.			
5.	Inhalte:	<p>Einleitung: Geschichte der Mikroprozessortechnik, Anwendungen, Mikroprozessoren und Mikrocontroller, Eingebettete Systeme</p> <p>Architektur eines Mikroprozessorsystems: Informationsverarbeitung, Zentraleinheit, Speicher, Peripheriemodule, von Neumann-Architektur vs. Harvard-Architektur, Programmiermodell, Befehlssatz, Befehlszyklus, Mikrocontroller</p> <p>Computerarithmetik: Vorzeichenlose und vorzeichenbehaftete Binärzahlen, Zweierkomplement-Darstellung, Arithmetisch-logische Operationen</p> <p>Befehlssatzarchitektur: Befehlssatzarchitektur eines 32-Bit RISC-Mikroprozessors, Befehlsformate, Adressierungsarten, Übersetzung von C-Programmkonstrukten in Assembly-Programmkonstrukte, Unterprogramme, Stapelspeicher, Rekursion, Werkzeuge zur Software-Entwicklung</p> <p>Mikroarchitektur: Logische Schaltungen, Rechenwerk, Steuerwerk, Register, Mikroarchitektur eines 32-Bit RISC-Mikroprozessors, Fließbandverarbeitung, Superskalare und VLIW-Mikroprozessoren</p> <p>Ausnahmebehandlung: Unterbrechungssystem, Identifikation und Priorisierung von Unterbrechungsquellen, vektorisierte Unterbrechungen, Ausnahmebehandlung</p>			

		<p>Speicher: Halbleiter-Speichertechnologien, Festwertspeicher ROM, statische / dynamische Schreib-Lese-Speicher SRAM / DRAM, Lokalitätsprinzip, Speicherhierarchie, Cache, virtueller Speicher</p> <p>Peripheriebausteine: Zähler und Zeitgeber, parallele Schnittstellen, serielle Schnittstellen, Direkter Speicherzugriff DMA, Analog/Digital-Wandler ADC</p> <p>8-Bit 8051-Prozessorfamilie: Mikrocontroller-Architektur, Programmiermodell, Befehlssatz, CPU-Zeitverhalten, Peripheriebausteine</p> <p>Praktikum: Programmierung und Implementierung von Assembler- und C-Programmen auf einem Mikroprozessor / Mikrocontroller (8051)</p> <p>Fachliteratur (Auswahl): [1] Neubauer, A.: Mikroprozessoren - Eine Einführung in die Befehlssatz- und Mikroarchitektur. Wilburgstetten: Schönbach Fachverlag, 2007 (ISBN 3-935340-56-4)</p>
6.	Teilnahmevoraussetzungen:	Digitaltechnik Informatik I (insbesondere Programmieren in C)
7.	Prüfungsformen:	Klausur
8.	Voraussetzungen für die Vergabe von Kreditpunkten:	- An- und Abtestate der Praktikumsaufgaben - Bestehen der Prüfung
9.	Häufigkeit des Angebots:	Jährlich
10.	Modulbeauftragter: Hauptamtlich Lehrende: Lehrbeauftragte:	Prof. Dr. André Neubauer Prof. Dr. André Neubauer —
11.	Sonstige Informationen:	Das Buch [1] entspricht der Vorlesung „Mikroprozessortechnik“. Weitere Informationen zu Vorlesung, Übung und Praktikum stehen den Studierenden auf dem ILIAS-Server zur Verfügung.

## 2.4 Projekt Softwareentwicklung

### 2.4.1 Modul Projekt Softwareentwicklung I

Kennnummer:	Aufwand:	Leistungs- punkte:	Studiense- mester:	Dauer:
L IT 2.4.1	120 h	4 LP	3.	1 Semester
1.	Lehrveranstaltung(en): Projekt Softwareentwicklung I	Kontaktzeit: 0+0+2 SWS	Selbststudium: 88 h	LP:
2.	Lehrformen:	Vorlesung+Übung+Praktikum: 0+0+2 SWS		
3.	Gruppengröße:	Vorlesung: 0, Übung: 0, Praktikum: ca. 3 x 10		
4.	Qualifikationsziele:	Die Studierenden kennen die Methoden der Projektplanung, -durchführung und -verfolgung. Sie verfügen über Methodenkompetenz zur fachbezogenen Erarbeitung neuer Themen, Kommunikation im Projektteam und Projektpräsentation.		
5.	Inhalte:	<p>Die Veranstaltung findet parallel zum Modul Informatik I statt und stützt sich auf die darin vermittelten Kenntnisse der Programmiersprache C und vertieft diese. Darüber hinaus werden folgende Inhalte einbezogen:</p> <p>Vermittlung von Grundlagen des Projektmanagements von Softwareentwicklungsprojekten (Teil 1).</p> <p>Vermittlung von Grundlagen der benötigten Präsentationstechniken.</p> <p>Einführung in die verwendeten Tools zur Versionsverwaltung (CVS, Subversion).</p> <p>Einführung in die Projektaufgabenstellung</p> <p>Erstellung eines Meilensteinplans, Einteilung der Teams und Zuweisung von Teilaufgaben an Teams</p> <p>Praktikum: Beispielhaftes Projekt: Entwicklung eines Simulators für zelluläre Automaten; Berechnung und Ausgabe einer animierten Grafik, die eine Game-of-Life-Simulation (nach John Conway) darstellt.</p> <p>Fachliteratur (Auswahl): [1] H.-P. Gumm, M. Sommer: Einführung in die Informatik, Oldenbourg, 5. Auflage, 2006 [2] R. Sedgewick: Algorithmen in C, Addison-Wesley, 1. Auflage, 1998. [3] B. W. Kernighan, D. M. Ritchie: The C Programming Language, Prentice Hall, Second Edition, 1988</p>		
6.	Teilnahmevoraussetzungen:	Vorkenntnisse in C oder parallele Belegung des Moduls Informatik I.		
7.	Prüfungsformen:	Besondere Prüfungsform (Präsentationen) / mündliche Prüfung		
8.	Voraussetzungen für die Vergabe von Kreditpunkten:	Abtestate der Teilaufgaben, Abschlusspräsentation und Einhaltung der Zieltermine sind Voraussetzung für die Zulassung zur Prüfung. Vergabe der Kreditpunkte nach Bestehen der Prüfung		
9.	Häufigkeit des Angebots:	Jährlich		

10.	Modulbeauftragter: Hauptamtlich Lehrende: Lehrbeauftragte:	Prof. Dr.-Ing. Ulrich Greveler Prof. Dr.-Ing. Ulrich Greveler —
11.	Sonstige Informationen:	



### 2.4.2 Modul Projekt Softwareentwicklung II

Kennnummer:	Aufwand:	Leistungs- punkte:	Studiense- mester:	Dauer:
L IT 2.4.2	120 h	4 LP	4.	1 Semester
1.	Lehrveranstaltung(en): Projekt Softwareentwicklung II	Kontaktzeit: 0+0+2 SWS	Selbststudium: 88 h	LP:
2.	Lehrformen:	Vorlesung+Übung+Praktikum: 0+0+2 SWS		
3.	Gruppengröße:	Vorlesung: 0, Übung: 0, Praktikum: ca. 3 x 10		
4.	Qualifikationsziele:	Die Studierenden kennen die Methoden der Projektplanung, -durchführung und -verfolgung. Sie verfügen über Methodenkompetenz zur fachbezogenen Erarbeitung neuer Themen, Kommunikation im Projektteam und Projektpräsentation.		
5.	Inhalte:	<p>Die Veranstaltung findet parallel zum Modul Informatik II statt und stützt sich auf die darin vermittelten Kenntnisse der Programmiersprache Java und vertieft diese. Es werden darüber hinaus folgende Themen behandelt:</p> <p>Einführung in die Grundlagen des Projektmanagements von Softwareentwicklungsprojekten (Teil 2)</p> <p>Einführung in die verwendeten Tools zur Versionsverwaltung und zum Testmanagement</p> <p>Erstellung von Testfällen und Abnahmeprozeduren</p> <p>Einführung in die Projektaufgabenstellung</p> <p>Erstellung eines Meilensteinplans, Einteilung der Teams und Zuweisung von Teilaufgaben an Teams</p> <p>Praktikum: Beispielhafte Projekte:</p> <p>Konzeption und Implementierung einer Client-Server-Architektur zur Aggregation und Verteilung von Rechenkapazität; Erstellung eines Clients zur Nutzung der bereitgestellten Rechenkapazität (z. B. Berechnung fraktaler Grafiken).</p> <p>Erstellung eines datenbankbasierten Webportals zur internen Kommunikation einer Gruppe mit spezifischen Anforderungen (z. B. Nutzerauthentisierung, Profilverwaltung, Suchabfragen)</p>		

		Fachliteratur (Auswahl): [1] H.-P. Gumm, M. Sommer: Einführung in die Informatik, Oldenbourg, 6. Auflage, 2004. [2] Barnhes, Kölling: Objektorientierte Programmierung mit Java, Pearson SStudio, 1. Auflage 2003. [3] H. Balzert: Lehrbuch der Software-Technik: Software-Entwicklung, Spektrum, 2. Auflage, 2000. [4] A. Heuer, G. Saake, K.-U. Sattler: Datenbanken kompakt, International Thomson Publishing, 1. Auflage 2001
6.	Teilnahmevoraussetzungen:	Erforderlich sind Vorkenntnisse in Java oder die parallele Belegung des Moduls Informatik II. Empfohlen wird die Teilnahme am Projekt Softwareentwicklung I.
7.	Prüfungsformen:	Besondere Prüfungsform (Präsentationen) / mündliche Prüfung
8.	Voraussetzungen für die Vergabe von Kreditpunkten:	Abtestate der Teilaufgaben, Abschlusspräsentation und Einhaltung der Zieltermine sind Voraussetzung für die Zulassung zur Prüfung. Vergabe der Kreditpunkte nach Bestehen der Prüfung
9.	Häufigkeit des Angebots:	Jährlich
10.	Modulbeauftragter: Hauptamtlich Lehrende: Lehrbeauftragte:	Prof. Dr.-Ing. Ulrich Greveler Prof. Dr.-Ing. Ulrich Greveler —
11.	Sonstige Informationen:	

## 2.5 Modul Software Engineering

Kennnummer:	Aufwand:	Leistungs- punkte:	Studiense- mester:	Dauer:
L IT 2.5	210 h	7 LP	5.	1 Semester
1.	Lehrveranstaltung(en): Software Engineering	Kontaktzeit: 2+0+3 SWS	Selbststudium: 130 h	LP:
2.	Lehrformen:	Vorlesung+Übung+Praktikum: 2+0+3 SWS		
3.	Gruppengröße:	Vorlesung: ca. 36, Übung: 0, Praktikum: ca. 3 x 12		
4.	Qualifikationsziele:	<p>Die Studierenden wissen, dass die arbeitsteilige Entwicklung komplexer Softwaresysteme sowohl in technischer als auch in organisatorischer Hinsicht einer ingenieurmäßigen Herangehensweise bedarf. Sie kennen die verschiedenen Phasen, in die sich der Lebenszyklus einer Software untergliedert, und sind mit unterschiedlichen Vorgehensmodellen für die zeitliche Abfolge und Wechselwirkung dieser Phasen vertraut.</p> <p>Die Studierenden kennen die wichtigsten Methoden und Diagrammformen für die Analyse der Anforderungen an ein Softwareprodukt. Sie kennen die wesentlichen Vorgehensweisen in den Softwareentwicklungsphasen Entwurf, Implementierung, Test und Betrieb. Die Studierenden wissen, welche spezifischen Praktiken das Management von Softwareprojekten ausmachen.</p>		
5.	Inhalte:	<p>Einführung: Definitionen für „Software“, Merkmale von Software gegenüber anderen technischen Produkten, (Wandel in den) Anforderungen an Software, Definitionen für „Software Engineering“, Stellenwert des Software Engineering innerhalb der Informatik</p> <p>Der Software-Lebenszyklus: Lebenszyklusphasen (Planung, Analyse, Entwurf, Implementierung, Test, Betrieb), Lebenszyklusmodelle (Code and fix, Wasserfall-Modell, V-Modell, Iterativ-inkrementelles Modell, Unified Process, Agilität, Extreme Programming)</p> <p>Analyse (Requirements Engineering): Anforderungen an ein Softwareprodukt (Klassifikation, Qualität, Ermittlung, Dokumentation), Objektorientierte Analyse und UML (Struktur- und Verhaltensmodellierung, Geschäftsprozess, Use-Case-Diagramm, Klassendiagramm, Aktivitätsdiagramm, Sequenzdiagramm, Zustandsautomat), Pflichtenheft</p> <p>Weitere Softwareentwicklungsphasen: Entwurf (Objektorientiertes Design, Entwurfsmuster, Datenhaltung, Systemarchitektur, Gestaltung von Benutzeroberflächen), Implementierung, Test (Fehlerursachen, Testmethoden, Testdurchführung und -dokumentation), Betrieb (Inbetriebnahme, Wartung, Pflege)</p>		

		<p>IT-Projektmanagement: Aufwandsschätzung (Ergebnis- und abwicklungsbezogene Einflussfaktoren, Vergleichsmethode, Algorithmische Methoden, Kennzahlenmodelle, Function-Point-Verfahren), IT-Spezifika der Projektplanung und -steuerung, Konfigurations- und Changemanagement</p> <p>Praktikum: Die Studierenden entwickeln im Rahmen eines Projektes, das sich über das gesamte Semester erstreckt, in Teamarbeit ein umfangreicheres Softwareprodukt. Dabei wenden sie in der Vorlesung erlernte Prinzipien und Methoden des Software Engineering praktisch an. Teil der Aufgabe ist auch die schrittweise Dokumentation und abschließende Präsentation der Entwicklungsergebnisse.</p> <p>Fachliteratur (Auswahl): [1] Zuser, Grechenig, Köhle: Software Engineering mit UML und dem Unified Process, Pearson 2004 [2] Balzert: Lehrbuch der Software-Technik, 2 Bände, Spektrum 2001 [3] Rupp: Requirements-Engineering und -Management, Hanser 2004</p>
6.	Teilnahmevoraussetzungen:	Das Modul baut auf die Veranstaltungen Informatik I-III, Objektorientierte Systeme und Datenbanken auf. Gute Programmierkenntnisse sind notwendig.
7.	Prüfungsformen:	Klausur, mündliche Prüfung oder besondere Prüfungsform
8.	Vorraussetzungen für die Vergabe von Kreditpunkten:	- Ausführlicher Projektstatusbericht zu jedem Praktikumstermin; Auslieferung, Präsentation und Dokumentation des vollständig fertiggestellten Softwareproduktes am letzten Praktikumstermin - Bestehen der Prüfung
9.	Häufigkeit des Angebots:	Jährlich
10.	Modulbeauftragter: Hauptamtlich Lehrende: Lehrbeauftragte:	Prof. Dr. Gernot Bauer Prof. Dr. Gernot Bauer —
11.	Sonstige Informationen:	

## 3 Wahlmodule

### 3.1 Modul Bussysteme

Kennnummer:		Aufwand:	Leistungs- punkte:	Studiense- mester:	Dauer:
L IT 3.1		210 h	7 LP	4.	1 Semester
1.	Lehrveranstaltung(en): Bussysteme		Kontaktzeit: 2+0+2 SWS	Selbststudium: 146 h	LP:
2.	Lehrformen:	Vorlesung+Übung+Praktikum: 2+0+2 SWS			
3.	Gruppengröße:	Vorlesung: ca. 72, Übung: 0, Praktikum: ca. 6 x 12			
4.	Qualifikationsziele:	Die Studierenden kennen die Prinzipien der Datenkommunikation in lokalen Netzen und den Aufbau moderner Daten-Netzwerke. Sie sind damit in der Lage, Bussysteme unter besonderer Berücksichtigung von Systemen der Automatisierungstechnik zu beurteilen, zu entwickeln und zu realisieren.			
5.	Inhalte:	<p>Physikalische Grundlagen: Signale, Signaldarstellung, Signalübertragung und Abtastung, Theorie lineare, zeitinvariante Systeme, Abtasttheorem, Modulation und Demodulation, Signalübertragung</p> <p>Datenübertragung: Signalintegrität (SI), Zeitanalyse, Übertragungsleitungen, Terminierung, Effekte der Signalintegrität, Übersprechen, Symbol Interferenzen , Nicht-idealer Rückleiter, Gleichzeitig schaltende Ausgänge, Taktverteilung</p> <p>Netzwerktopologie: Topologieklassifikation: statisch, dynamisch und hierarchisch, Segmente und Anschlusspriorität , Stichleitungen und Abzweigungen, Verdrahtungsebenen, Verbindungsebenen, Leitungslängen, Maximale Kopplungslängen, Leitungsimpedanzen</p> <p>Schnittstellen und Busstrukturen: Einführung High-Speed I/O Design, High-Speed I/O Design , Gehäuse, High-Speed Ausgangstreiber, Kompensation von I/O Treiber, High-Speed Eingangstreiber, I/O Signalprotokolle, I/O DFT und DFM, Taktverteilung auf Systemebene</p> <p>Systembusse: Überblick über serielle Schnittstellen, Takt- und Datenrückgewinnung, Überträgerdesign, Empfängerdesign, Beispiele: PCI-E, USB/ wireless USB, SATA, FSB</p>			

		<p>LAN, MAN, WAN: Modelle, Bauteile, LAN, MAN, WAN</p> <p>Praktikum: Einführung in SPICE, Rechnergestützter Schaltungsentwurf (AMS), Transmission Line Design, Einführung in wireless USB, iDwaRF Netzwerk-Protokoll</p> <p>Fachliteratur (Auswahl): [1] R. J. Baker, CMOS Circuit Design, Layout and Simulation (IEEE Press Series on Microelectronic Systems), 2004 [2] Digital Integrated Circuits: A Design Perspective, Prentice Hall 2003 [3] ITRS, International Technology Roadmap for Semiconductors, www.itrs.net</p>
6.	Teilnahmevoraussetzungen:	Grundkenntnisse in Informatik und Elektrotechnik
7.	Prüfungsformen:	Klausur oder mündliche Prüfung
8.	Voraussetzungen für die Vergabe von Kreditpunkten:	- An- und Abtestate der Praktikumsaufgaben - Bestehen der Prüfung
9.	Häufigkeit des Angebots:	Jährlich
10.	Modulbeauftragter: Hauptamtlich Lehrende: Lehrbeauftragte:	Prof. Dr. Peter Glösekötter Prof. Dr. Peter Glösekötter —
11.	Sonstige Informationen:	

## 3.2 Modul Daten- und Netzwerksicherheit

Kennnummer:	Aufwand:	Leistungs- punkte:	Studiense- mester:	Dauer:
L IT 3.2	210 h	7 LP	5.	1 Semester
1.	Lehrveranstaltung(en): Daten- und Netzwerksicherheit	Kontaktzeit: 2+0+2 SWS	Selbststudium: 146 h	LP:
2.	Lehrformen:	Vorlesung+Übung+Praktikum: 2+0+2 SWS		
3.	Gruppengröße:	Vorlesung: ca. 36, Übung: 0, Praktikum: ca. 3 x 12		
4.	Qualifikationsziele:	<p>Die Studierenden kennen grundlegende Bedrohungen von informationstechnischen Systemen und zugehörige Anforderungen an IT-Sicherheit sowie die Grundlagen des technischen Datenschutzes. Zentrale kryptographische Mechanismen, Protokolle und ihre Parameter sind bekannt.</p> <p>Die Studierenden sind in der Lage, Bedrohungen einer unternehmensweiten IT-Landschaft zu erkennen und diesen unter Einleitung konkreter Maßnahmen entgegenzuwirken, Anforderungen umzusetzen und sicherheitstechnische Werkzeuge in IT-Projekten einzusetzen.</p>		
5.	Inhalte:	<p><b>Begriffswelt der IT-Sicherheit:</b> Vertraulichkeit, Integrität, Authentisierung, Verfügbarkeit, Angriffsmodelle, Risiko, Datenschutz und -sicherheit, vertrauenswürdige Komponenten, kryptographische Algorithmen und Schlüssel, Virus, Wurm, Trojanisches Pferd, Rootkit.</p> <p><b>Mechanismen:</b> Verschlüsselung, Symmetrische Verfahren, Stromchiffren, Blockchiffren, asymmetrische Verfahren, Hashfunktionen, diskreter Logarithmus, Langzahlarithmetik, RSA, Diffie-Hellmann, ElGamal; digitale Signatur, Zertifikate, PKI.</p> <p><b>Sichere Netze:</b> Firewalls, Intrusion Prevention, DMZ, Virtual Private Network, Remote Access Service und Authentifizierungsprotokolle (z. B. RADIUS, Kerberos), Absicherung von Funknetzen (z. B. WLAN).</p> <p>Einführung in das Thema Evaluation und Zertifizierung nach formalen Sicherheitskriterien, Audits, Grundschutz nach BSI.</p> <p><b>Praktikum:</b> Das Praktikum orientiert sich an den Inhalten der Vorlesung.</p> <p><b>Fachliteratur (Auswahl):</b> [1] A. Beutelspacher, H.B. Neumann, T.Schwarzpaul, Kryptografie in Theorie und Praxis, Vieweg, Wiesbaden, 2005 [2] William R. Cheswick, Steven M. Bellovin und Aviel D. Rubin, Firewalls und Sicherheit im Internet, Addison-Wesley, München; 2004 [3] Claudia Eckert, IT-Sicherheit - Konzepte - Verfahren - Protokolle, R. Oldenbourg Verlag, 2004 [4] Jörg Schwenk, Sicherheit und Kryptographie im Internet, Vieweg,</p>		

6.	Teilnahmevoraussetzungen:	Bestandene Module Informatik I, Informatik II, Rechnerarchitektur und Betriebssysteme. Begleitend soll zudem die Veranstaltung Netzwerkprogrammierung belegt werden.
7.	Prüfungsformen:	Klausur oder mündliche Prüfung
8.	Voraussetzungen für die Vergabe von Kreditpunkten:	- An- und Abtestate der Praktikumsaufgaben - Bestehen der Prüfung
9.	Häufigkeit des Angebots:	Jährlich
10.	Modulbeauftragter: Hauptamtlich Lehrende: Lehrbeauftragte:	Prof. Dr. Ulrich Greveler Prof. Dr. Ulrich Greveler —
11.	Sonstige Informationen:	Vorlesungsunterlagen sind elektronisch verfügbar URL: <a href="http://www.its.fh-muenster.de">http://www.its.fh-muenster.de</a>

### 3.3 Modul Kommunikationssysteme

Kennnummer:		Aufwand:	Leistungs- punkte:	Studiense- mester:	Dauer:
L IT 3.3		210 h	7 LP	4.	1 Semester
1.	Lehrveranstaltung(en): Kommunikationssysteme	Kontaktzeit: 3+1+1 SWS		Selbststudium: 130 h	LP:
2.	Lehrformen:	Vorlesung+Übung+Praktikum: 3+1+2 SWS			
3.	Gruppengröße:	Vorlesung: ca. 36, Übung: ca. 36, Praktikum: ca. 3 x 12			
4.	Qualifikationsziele:	Die Studierenden kennen auf der Basis der leitungsgebundenen und funkgestützten Kommunikationstechnik moderne Systeme der Sprach- und Datenkommunikation. Sie sind dadurch in der Lage, weitergehende Analysen zukünftiger Daten- und Sprachsysteme mit zu entwickeln.			
5.	Inhalte:	<p>Geschichte: Entwicklungsgeschichte der Telekommunikation, Information, Kommunikation und Wissen, Benutzeranforderungen</p> <p>Grundlagen: Telekommunikation, Netze und Dienste, Netzbetreiber, Netztopologie, Betriebsarten, Übertragungsverfahren, Übertragungsmedium, Vielfachzugriffsverfahren, Vermittlungstechnik, Kommunikationsrichtung und -art, Informationstyp</p> <p>Kommunikationsmodell: Schichtenbildung, OSI-Schichtenmodell, Bitübertragungsschicht, Sicherungsschicht, Vermittlungsschicht, Transportschicht, Sitzungsschicht, Darstellungsschicht, Anwendungsschicht</p> <p>Analoge Sprachkommunikation: Telefonnetz, Intelligentes Netz, Telefondienste, Telex, Telefax, Datenfernübertragung, T-Online</p> <p>Datenkommunikation: Datennetze, Datex-P, Datex-L, Datex-M, Datendirektverbindungen, Lokale Rechnernetze, Weltweite Rechnernetze, Verbundleistungen</p> <p>Verkehrstheorie: Warteschlangenmodell, Erlang, Busy Hour, Poissonverteilung, Netzzusammenschaltungen</p> <p>Informationstheorie und -codierung: Zufall, Wahrscheinlichkeitslehre, Quelle, Kanal, Abtastung und Quantisierung, Quellencodierung, Kanalcodierung, Leitungscodierung</p>			

		<p>Digitale Telefonnetze und Vermittlungstechnik: Leistungsmerkmale, Signalisierung, Verbindungsaufbau, Basisanschluss, Primärmultiplexanschluss, PCM 30-System, Koppelnetze, Zeitlagenvielfach, Raumlagenvielfach</p> <p>Mobilfunk: Netzaufbau, Systemfunktion, Sprachcodec, Fehlerschutzmechanismen, Sicherheitsaspekte</p> <p>Praktikum: Infrarot-Technologie von Fernbedienungen, LIRC-Empfänger unter Linux, ISDN, PCM-Strecken, Firewall unter Linux, DECT-System</p> <p>Fachliteratur (Auswahl): [1] Eberspächer, Jörg; Vögel, Hans-Jörg; Brettstetter, Christian: GSM Global System for Mobile Communication. Stuttgart: Teubner Verlag, 2005. ISBN 3-519-26192-8 [2] Nocker, Rudolf: Digitale Kommunikationssysteme. Bd. 1 &amp; 2. Wiesbaden: Vieweg Verlag, 2005. ISBN 3-528-03976-0 und 3-528-03977-9 [3] Schiller, Jochen: Mobilkommunikation. Techniken für das allgegenwertige Internet. München: Pearson Studium (Addison-Wesley), 2003. ISBN 3-8273-7060-4 [4] Weidenfeller, Hermann: Grundlagen der Kommunikationstechnik. Stuttgart: Teubner Verlag, 2002. ISBN 3-519-06265-8</p>
6.	Teilnahmevoraussetzungen:	Grundlagen der Elektrotechnik, Digitaltechnik
7.	Prüfungsformen:	Klausur oder mündliche Prüfung
8.	Voraussetzungen für die Vergabe von Kreditpunkten:	- An- und Abtestate der Praktikumsaufgaben - Bestehen der Prüfung
9.	Häufigkeit des Angebots:	Jährlich
10.	Modulbeauftragter: Hauptamtlich Lehrende: Lehrbeauftragte:	Prof. Dr. Peter Richert Prof. Dr. Peter Richert —
11.	Sonstige Informationen:	Das Skript zur Vorlesung steht unter <a href="http://www.ktet.fh-muenster.de/Download/Studium/KT">http://www.ktet.fh-muenster.de/Download/Studium/KT</a> zur Verfügung.

## 3.4 Modul Local Area Networks

Kennnummer:	Aufwand:	Leistungs- punkte:	Studiense- mester:	Dauer:
L IT 3.4	210 h	7 LP	4.	1 Semester
1.	Lehrveranstaltung(en): Local Area Networks	Kontaktzeit: 2+1+1 SWS	Selbststudium: 146 h	LP:
2.	Lehrformen:	Vorlesung+Übung+Praktikum: 2+1+1 SWS		
3.	Gruppengröße:	Vorlesung: ca. 36, Übung: ca. 36, Praktikum: ca. 3 x 12		
4.	Qualifikationsziele:	Die Studierenden kennen die wesentlichen Protokolle der IP-Protokollfamilie und verstehen die Funktionsweise IP-basierter Netze. Sie sind damit in der Lage, Designprinzipien von Protokollen zu beurteilen und Entscheidungen zur Wahl eines Transportprotokolls für Anwendungen zu treffen.		
5.	Inhalte:	<p>Netztopologien</p> <p>Das OSI- und das Internet Modell</p> <p>Verschiedene Linklayer</p> <p>Ethernetbasierte Netze, ARP, Virtuelle LANs, Trunking, Spanning Tree, PPPoE</p> <p>Netzwerkprotokolle IPv4 und IPv6</p> <p>Hilfsprotokolle ICMP und ICMPv6</p> <p>Transportprotokolle UDP, TCP und SCTP</p> <p>Überlast- und Flusskontrollmechanismen bei TCP und SCTP</p> <p>Link State und Distance Vector Routing</p> <p>Network Address Translation</p> <p>Sicherheitsaspekte, Transport Layer Security, Firewalls; SSH, Port Forwarding</p> <p>Beispiele von Anwendungsprotokollen wie SMTP, HTTP, POP-3, FTP</p> <p>Praktikum: Nutzung des Paketanalysators Wireshark; Analyse der Flowcontrol und Silly Window Syndrome Avoidance bei TCP; Analyse von Traceroute; SSH mit Port Forwarding; IPv4 in IPv4 Tunnel und IPv6 in IPv4 Tunnel; Emulation von Netzwerkparametern und Routing.</p>		

		<p>Fachliteratur (Auswahl):</p> <p>[1] Richard W. Stevens: TCP/IP Illustrated, Volume I, Addison-Wesley, 1st Edition, 1993.</p> <p>[2] Radia Perlmann: Interconnections - Bridges, Routers, Switches, and Internetworking Protocols, Addison-Wesley, 2nd Edition, 1999.</p> <p>[3] Behrouz A. Forouzan: TCP/IP Protocol Suite, McGraw-Hill, 2nd Edition, 2002.</p> <p>[4] Andrew S. Tanenbaum: Computer Networks, 3rd Edition, Prentice Hall.</p>
6.	Teilnahmevoraussetzungen:	Grundlegende Kenntnisse der Informatik im Umfang der Module Informatik I und Informatik II.
7.	Prüfungsformen:	Klausur oder mündliche Prüfung
8.	Voraussetzungen für die Vergabe von Kreditpunkten:	- An- und Abstate der Praktikumsaufgaben - Bestehen der Prüfung
9.	Häufigkeit des Angebots:	Jährlich
10.	Modulbeauftragter: Hauptamtlich Lehrende: Lehrbeauftragte:	Prof. Dr. Michael Tüxen Prof. Dr. Michael Tüxen —
11.	Sonstige Informationen:	

## 3.5 Mathematik

### 3.5.1 Modul Mathematik III für Informatiker

Kennnummer:		Aufwand:	Leistungs- punkte:	Studiense- mester:	Dauer:
L IT 3.5.1		210 h	7 LP	5.	1 Semester
1.	Lehrveranstaltung(en): Mathematik III für Informatiker		Kontaktzeit: 4+2+0 SWS	Selbststudium: 114 SWS	LP:
2.	Lehrformen:	Vorlesung+Übung+Praktikum: 4+2+0 SWS			
3.	Gruppengröße:	Vorlesung: ca. 30, Übung: ca. 30, Praktikum: 0			
4.	Qualifikationsziele:	Die Studierenden beherrschen die grundlegenden Begriffe und Methoden der höheren Mathematik in den Teilgebieten Zahlentheorie, Algebra, Wahrscheinlichkeitstheorie und Graphentheorie. Sie verfügen über die Kompetenz zur Anwendung der mathematischen Begriffe und Methoden in weiterführenden Veranstaltungen der Informatik.			
5.	Inhalte:	<p>Zahlentheorie: Arithmetik in <math>\mathbb{Z}</math>, Teiler, Primfaktorzerlegung, Phi-Funktion, Satz von Euler, RSA Algorithmus</p> <p>Algebra: Äquivalenzrelationen, Quotienten, Gruppen, Ringe, Körper, Moduln, <math>\mathbb{Z}/n\mathbb{Z}</math>, lineare Algebra über <math>\mathbb{Z}/2\mathbb{Z}</math></p> <p>Wahrscheinlichkeitstheorie: Zufallsexperimente, Wahrscheinlichkeitsbegriff, bedingte Wahrscheinlichkeit, stochastische Unabhängigkeit, Bayes'sche Formel, Zufallsvariable, Wahrscheinlichkeitsverteilungen, Erwartungswert, Varianz, Binomialverteilung, Poisson-Verteilung, Normalverteilung, zentraler Grenzwertsatz, Stochastische Prozesse, Warteschlangen</p> <p>Graphentheorie: Grundbegriffe, Wege, Kreise, Zusammenhang, Bäume, Eulersche Graphen, Hamiltonsche Graphen, Wälder, kürzeste Wege, Flüsse, Färbungen</p> <p>Fachliteratur (Auswahl): [1] A. Steger, T. Schickinger, Diskrete Strukturen, Springer Verlag, 2002 [2] K.-U. Witt, Algebraische Grundlagen der Informatik, Vieweg Verlag, 2005 [3] K. Denecke, Algebra und diskrete Mathematik für Informatiker, Teubner, 2003 [4] G. Hübner: Stochastik, Vieweg Verlag, 1995. [5] R. Nelson: Probability, Stochastic Processes and Queueing Theory, Springer Verlag, 1995.</p>			
6.	Teilnahmevoraussetzungen:	Kenntnisse aus den Modulen Mathematik I und Mathematik II			
7.	Prüfungsformen:	Klausur oder mündliche Prüfung			
8.	Vorraussetzungen für die Vergabe von Kreditpunkten:	Bestehen der Prüfung			

9.	Häufigkeit des Angebots:	Jährlich
10.	Modulbeauftragter: Hauptamtlich Lehrende: Lehrbeauftragte:	Prof. Dr. Gernot Bauer Prof. Dr. Gernot Bauer Prof. Dr. Hans Effinger —
11.	Sonstige Informationen:	



## 3.6 Modul Netzwerkprogrammierung

Kennnummer:		Aufwand:	Leistungs- punkte:	Studiense- mester:	Dauer:
L IT 3.6		210 h	7 LP	5.	1 Semester
1.	Lehrveranstaltung(en): Netzwerkprogrammierung	Kontaktzeit: 2+0+2 SWS		Selbststudium: 146 h	LP:
2.	Lehrformen:	Vorlesung+Übung+Praktikum: 2+0+2 SWS			
3.	Gruppengröße:	Vorlesung: ca. 36, Übung: 0, Praktikum: ca. 3 x 12			
4.	Qualifikationsziele:	Die Studierenden kennen die Architekturen von Netzwerkanwendungen und sind mit dem Socket API zur Nutzung der Transportprotokolle UDP, TCP und SCTP vertraut. Sie sind damit in der Lage, Anwendungsprotokolle zu spezifizieren und interoperabel zu implementieren.			
5.	Inhalte:	<p>Die Autotools</p> <p>Socketschnittstelle von UDP, TCP und SCTP</p> <p>Benutzung des DNS</p> <p>Protokollunabhängigkeit</p> <p>Serverarchitekturen: Beispiele einiger Server</p> <p>Signalbehandlung: Non-Blocking API, Benutzung von Uni-, Multi- und Broadcast</p> <p>Protokolldesign: Sicherheitsaspekte; Design eines Netzwerkspiels</p> <p>Praktikum: Implementierung einfacher Server (Discard, Echo, Daytime und Chargen) und eines Clients, die verschiedene Betriebssysteme unterstützen, sowohl auf Little Endian als auch auf Big Endian Prozessoren lauffähig sind und IPv4 und IPv6 unterstützen. Als Transportprotokolle werden UDP, TCP und SCTP genutzt. Es muss ein Netzwerkspiel interoperabel implementiert werden.</p> <p>Fachliteratur (Auswahl): [1] W. R. Stevens: Network Programming, Volume 1, 3rd Edition, Prentice Hall, 2003. [2] W. Richard Stevens: TCP/IP Illustrated, Volume 1, Addison Wesley, 1994. [3] M. Zahn: Unix-Netzwerkprogrammierung mit Threads, Sockets und SSL, Springer Verlag, 2006.</p>			
6.	Teilnahmevoraussetzungen:	Kenntnisse auf dem Gebiet die IP Protokolle im Umfang des Moduls Local Area Networks. Ferner Gute Programmierkenntnisse in C und elementare UNIX-Kenntnisse.			
7.	Prüfungsformen:	Klausur oder mündliche Prüfung			

8.	Vorraussetzungen für die Vergabe von Kreditpunkten:	- An- und Abtestate der Praktikumsaufgaben - Bestehen der Prüfung
9.	Häufigkeit des Angebots:	Jährlich
10.	Modulbeauftragter: Hauptamtlich Lehrende: Lehrbeauftragte:	Prof. Dr. Michael Tüxen Prof. Dr. Michael Tüxen —
11.	Sonstige Informationen:	



## 3.7 Modul Objektorientierte Systeme

Kennnummer:	Aufwand:	Leistungs- punkte:	Studiense- mester:	Dauer:
L IT 3.7	210 h	7 LP	4.	1 Semester
1.	Lehrveranstaltung(en): Objektorientierte Systeme	Kontaktzeit: 2+0+2 SWS	Selbststudium: 146 h	LP:
2.	Lehrformen:	Vorlesung+Übung+Praktikum: 2+0+2 SWS		
3.	Gruppengröße:	Vorlesung: ca. 36, Übung: 0, Praktikum: ca. 3 x 12		
4.	Qualifikationsziele:	Die Studierenden kennen die Konzeption und Implementierung von größeren Software-Systemen mit Hilfe von aktuellen Design und Architekturmustern. Der gesamte Projektlebenszyklus, angefangen von der Analyse bis hin zur Implementierung und Test, wird beherrscht.		
5.	Inhalte:	<p>Vorgehensmodelle zur Softwareerstellung</p> <p>Use Case Analyse und Design Modell mit der UML</p> <p>Design und Architektur Muster</p> <p>Muster für verteilte Anwendungen</p> <p>Software Architekturen und Frameworks</p> <p>Java Idioms und ihre Implementierung</p> <p>Software Qualität</p> <p>Praktikum: Erstellung einer verteilten Anwendung unter Verwendung von Design und Architektur Mustern.</p> <p>Fachliteratur (Auswahl):            [1] J. Rumbaugh et. al.: Object-Oriented Modeling and Design, Prentice Hall, 1991.            [2] G. Booch: Object-Oriented Analysis and Design with Applications, Benjamin/Cummings, 1994.            [3] E. Gamma et. al.: Design Patterns, Addison-Wesley, 1995.            [4] F. Buschmann et al.: Pattern-orientierte Software-Architektur, Addison-Wesley, 1998.</p>		
6.	Teilnahmevoraussetzungen:	Informatik I und II, fundierte Programmierkenntnisse in Java.		
7.	Prüfungsformen:	Klausur, mündliche Prüfung oder besondere Prüfungsform		
8.	Vorraussetzungen für die Vergabe von Kreditpunkten:	- An- und Abtestate der Praktikumsaufgaben - Bestehen der Prüfung		
9.	Häufigkeit des Angebots:	Jährlich		
10.	Modulbeauftragter: Hauptamtlich Lehrende: Lehrbeauftragte:	Prof. Dr. Nikolaus Wulff Prof. Dr. Nikolaus Wulff —		
11.	Sonstige Informationen:	Das Skript zur Vorlesung steht unter <a href="http://www.lab4inf.fh-muenster.de">http://www.lab4inf.fh-muenster.de</a> zur Verfügung.		



## 3.8 Modul Projektmanagement

Kennnummer:	Aufwand:	Leistungs- punkte:	Studiense- mester:	Dauer:
L IT 3.8	120 h	4 LP	5.	1 Semester
1.	Lehrveranstaltung(en): Projektmanagement	Kontaktzeit: 1+2+1 SWS	Selbststudium: 56 h	LP:
2.	Lehrformen:	Vorlesung+Übung+Praktikum: 1+2+1 SWS		
3.	Gruppengröße:	Vorlesung: ca. 36, Übung: ca. ca. 6 x 6, Praktikum: ca. 6 x 6		
4.	Qualifikationsziele:	Die Studierenden kennen die wesentlichen Grundlagen des Projektmanagements und wenden diese Kenntnisse an einem realen Projekt an. Sie sind in der Lage, Projekte zu planen, durchzuführen, in Teilen zu leiten sowie Projektfortschritt und -ergebnisse zu dokumentieren und zu präsentieren.		
5.	Inhalte:	<p>Grundlagen des Projektmanagements</p> <p>Was ist ein Projekt</p> <p>Wann ist Projektmanagement sinnvoll</p> <p>Ganzheitliches Projektmanagement</p> <p>Merkmale eines Projektes</p> <p>Projektvereinbarung, Projektkosten, Erfolgreiche Projektarbeit,</p> <p>Projektauftrag</p> <p>Umfeldanalyse, Mitarbeiter, Projektstrukturen, Meilensteine, Projektstrukturplan, Widerstände im Projekt</p> <p>Projekt-Abschluss</p> <p>Praktikum:          Bearbeiten eines konkreten Projekts nach Absprache mit den betreuenden Hochschullehrern          Beispiele für Projektarbeiten: (1) Einrichtung eines Multimedia-Arbeitsplatzes incl., (2) Installation von Messgeräten, (3) Studie zur Realisierung eines DECT-Funkmoduls, (4) Aufbau einer Schrittmotor-gesteuerten Antenne für 10,4 Ghz, (5) Entwicklung einer Aarmanlage mit Multi-Sensor-Eingängen und GSM-Signalisierung, (6) Entwicklung eines Spektrum-Analysators für 144 Mhz, (7) Untersuchungen zu Protokollen für Voice-over-IP, (8) Grundlegende Untersuchungen zum Bürger-LAN, (9) Simulation einer optischen Lageregelung</p> <p>Fachliteratur (Auswahl):          [1] H.-D. Litke, I. Kunow, Projektmanagement, Haufe-Verlag          [2] W. Lessel, Projektmanagement, Cornelsen-Verlag</p>		

6.	Teilnahmevoraussetzungen:	Grundlagen der Elektrotechnik, Analogelektronik, Elektronische Bauelemente
7.	Prüfungsformen:	Klausur, mündliche Prüfung oder besondere Prüfungsform
8.	Voraussetzungen für die Vergabe von Kreditpunkten:	- Erfolgreiche Durchführung des Projekts - Projektpräsentation (Vortrag) - Bestehen der Prüfung
9.	Häufigkeit des Angebots:	Jährlich
10.	Modulbeauftragter: Hauptamtlich Lehrende: Lehrbeauftragte:	Prof. Dr. Dirk Fischer Alle Hochschullehrer des Fachbereichs —
11.	Sonstige Informationen:	

## 3.9 Modul Rechnerarchitektur

Kennnummer: L IT 3.9		Aufwand: 210 h 210 h	Leistungs- punkte: 7 LP 7 LP	Studiense- mester: 4. 5.	Dauer: 1 Semester 1 Semester
1.	Lehrveranstaltung(en): Rechnerarchitektur		Kontaktzeit: 2+0+1 SWS 2+0+1 SWS	Selbststudium: 98 h 98 h	LP: 7 LP 7 LP
2.	Lehrformen:	Vorlesung+Übung+Praktikum: 4+0+2 SWS			
3.	Gruppengröße:	Vorlesung: ca. 36, Übung: 0, Praktikum: ca. 3 x 12			
4.	Qualifikationsziele:	Die Studierenden kennen die Funktionsweise eines Rechnersystems und verstehen aktuelle Architekturansätze als Konsequenz von Leistungsvorgaben. Sie beherrschen die Prototypenentwicklung mit VHDL.			
5.	Inhalte:	<p>Einführung: Geschichte, Begriffe</p> <p>VHDL: Aufgabenbereich der Sprache: Pflichtenheft, Dokumentation, Simulation, Synthese. Syntax, Unterschiede zu klassischen Software-Sprachen: Nebenläufigkeit, Strukturunterstützung, sequenzielle Teile, explizites Timing. Start und Beendigung von Prozessen. Unterschiede zwischen SIGNAL und VARIABLE, zwischen WHEN, CASE und IF. Komponenteninstanziierung. Multiple Architekturen, Anlegen von Bibliotheken, Multi-level Simulation.</p> <p>Elemente eines modernen Rechnersystems: Rechenwerke, Leitwerke, CPU, Speicher, Elementare Maschinenbefehle für einfache Rechner. Architekturansätze zur Leistungssteigerung: Hauptspeicher und Cache, Pipelining &amp; Vektorrechner, Superskalarrechner. Instruktionssatz als Pendant zur Hardware, Pipelinebefehle &amp; VLIW, EPIC und Hyperthreading</p> <p>Konfigurierbare Hardware: Nicht-flüchtige Speicherung: Sicherung, Antifuse, Floating Gate. Konfigurierbare Logik: PLA, PLD &amp; FPGA. Schaltungssynthese. Hardware-Realisierung paralleler Prozesse. Bedienen von Schnittstellen. „Eigene Software für eigene Hardware“</p> <p>Praktikum: Elementarer VHDL Entwurf, Endliche Automaten, Umwandlung von Datenformaten, Schaltungssynthese, Konfigurieren von FPGAs, Realisierung von Hardwareschnittstellen, Entwurf eines Rechnerkerns auf einem FPGA und dessen Anwendung in komplexeren Systemen</p>			

		Fachliteratur (Auswahl): [1] C. Märtin, Rechnerarchitekturen, Carl Hanser Verlag, Leipzig, 2001, ISBN 3-446-21475-5 [2] G. Lehman, .Wunder & M. Selz: Schaltungsdesign mit VHDL, Franzis Verlag, Poing 1994, ISBN 3-7723-6163-3
6.	Teilnahmevoraussetzungen:	Inhaltlich baut das Modul auf die Veranstaltungen Digitaltechnik auf und greift auf Teile der Mikroprozessortechnik zurück.
7.	Prüfungsformen:	Klausur und/oder Prototypenprojekt
8.	Voraussetzungen für die Vergabe von Kreditpunkten:	- An- und Abtestate der Praktikumsaufgaben - Bestehen der Prüfung
9.	Häufigkeit des Angebots:	Jährlich
10.	Modulbeauftragter: Hauptamtlich Lehrende: Lehrbeauftragte:	Prof. Dr. Martin Poppe Prof. Dr. Martin Poppe —
11.	Sonstige Informationen:	