

# Modulhandbuch

---

## Bachelor

# Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik

---

**Studienordnungsversion: 2013**

**Vertiefung: IN**

**gültig für das Studiensemester 2016**

Erstellt am: 02. Mai 2016

aus der POS Datenbank der TU Ilmenau

Herausgeber: Der Rektor der Technischen Universität Ilmenau

URN: urn:nbn:de:gbv:ilm1-mhb-572

# Inhaltsverzeichnis

Name des Moduls/Fachs	1.FS	2.FS	3.FS	4.FS	5.FS	6.FS	7.FS	Abschluss	LP	Fachnr.
	VSP	VSP	VSP	VSP	VSP	VSP	VSP			
Informatik								FP	60	
Grundlagen und Diskrete Strukturen								FP	6	100541
GL u. Diskrete Strukturen			4 2 0					PL	6	100542
Programmierparadigmen und Kommunikationsmodelle								FP	7	100333
Kommunikationsmodelle				2 1 0				VL	3	255
Programmierparadigmen				2 2 0				VL	4	5378
Softwaretechnik								FP	6	100334
Softwaretechnik 1				2 1 0				PL 90min	3	100533
Softwaretechnik 2				2 1 0				PL 90min	3	100564
Telematik 1								FP	5	100322
Telematik 1				3 1 0				PL 90min	5	100575
Datenbank- und Betriebssysteme								FP	8	100335
Betriebssysteme				2 1 0				VL	4	252
Datenbanksysteme				2 1 0				VL	4	244
Algorithmen, Automaten und Komplexität								FP	6	100321
Algorithmen und Datenstrukturen 1				2 1 0				VL	3	100534
Automaten und Komplexität				2 1 0				VL	3	100535
Proseminar für IN Bsc								MO	2	100346
Proseminar für IN Bsc					0 2 0			SL	2	100956
Computergrafik								FP	5	100680
Computergrafik					3 1 0			PL 60min	5	5367
Logik und Logikprogrammierung								FP	5	100339
Logik und Logikprogrammierung						3 2 0		PL 150min	5	100574
Wahlpflichtmodul								FP	8	101210
Rechnerarchitekturen für IN								FP	8	100331
Rechnerarchitekturen 1				2 2 0				PL 90min	4	5382
Rechnerarchitekturen 2								PL 90min	3	5383
Praktikum Rechnerarchitekturen 1 und 2						0 0 1		SL	1	100561
Neuroinformatik und Schaltsysteme								FP	8	100332
Neuroinformatik								VL	3	1389
Schaltsysteme								VL	3	100457



---

## Modul: Informatik

Modulnummer101211

Modulverantwortlich: Prof. Dr. Martin Dietzfelbinger

Modulabschluss: Fachprüfung/Modulprüfung generiert

Lernergebnisse

Vorraussetzungen für die Teilnahme

Detailangaben zum Abschluss

---

## Modul: Grundlagen und Diskrete Strukturen

Modulnummer100541

Modulverantwortlich: Prof. Dr. Matthias Kriesell

Modulabschluss: Fachprüfung/Modulprüfung generiert

Lernergebnisse

siehe Fachbeschreibung

Vorraussetzungen für die Teilnahme

Detailangaben zum Abschluss

## GL u. Diskrete Strukturen

Fachabschluss: mehrere Teilleistungen  
 Sprache: Deutsch/Englisch

Art der Notengebung: Generierte Noten

Pflichtkennz.: Pflichtfach

Turnus: Wintersemester

Fachnummer: 100542

Prüfungsnummer: 240251

Fachverantwortlich: Prof. Dr. Matthias Kriesell

Leistungspunkte: 6	Workload (h): 180	Anteil Selbststudium (h): 112	SWS: 6.0
Fakultät für Mathematik und Naturwissenschaften			Fachgebiet: 2411

SWS nach Fachsemester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
							4	2	0												

### Lernergebnisse / Kompetenzen

Kenntnisse in Grundlagen der Mathematik, grundlegende mathematische Arbeitsweisen (Beweise)

### Vorkenntnisse

Abiturwissen Mathematik

### Inhalt

Logik und Mengenlehre; Funktionen und Relationen; Gruppen, Ringe, Körper; Boolesche Algebren; diskrete Wahrscheinlichkeitsräume; elementare Graphentheorie

### Medienformen

Tafel

### Literatur

Wird in der Vorlesung bekanntgegeben

### Detailangaben zum Abschluss

werden bei Bedarf festgelegt

### verwendet in folgenden Studiengängen

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Elektrotechnik 2013 Vertiefung MA

Bachelor Informatik 2013

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik 2013 Vertiefung IN

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik 2013 Vertiefung MA

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Elektrotechnik 2013 Vertiefung IN

---

## Modul: Programmierparadigmen und Kommunikationsmodelle

Modulnummer 100333

Modulverantwortlich: Prof. Dr. Winfried Kühnhauser

Modulabschluss: Fachprüfung/Modulprüfung generiert

### Lernergebnisse

Die Studierenden lernen in diesem Modul grundlegende Paradigmen und Modelle zur Programmierung von Softwaresystemen und ihrer Kommunikation kennen. Sie verstehen die methodischen Grundlagen moderner Programmiersprachen und Middleware-Plattformen und sind in der Lage, sie bezüglich ihrer Leistung in unterschiedlichen Anwendungsdomänen der Informatik zu analysieren und bewerten.

### Voraussetzungen für die Teilnahme

### Detailangaben zum Abschluss

schriftliche Modulprüfung aus zwei Teilen (120 min)

## Kommunikationsmodelle

Fachabschluss: über Komplexprüfung  
 Sprache: Deutsch

Art der Notengebung: unbenotet

Pflichtkennz.: Pflichtfach

Turnus: Wintersemester

Fachnummer: 255

Prüfungsnummer: 2200378

Fachverantwortlich: Prof. Dr. Winfried Kühnhauser

Leistungspunkte: 3      Workload (h): 90      Anteil Selbststudium (h): 56      SWS: 3.0  
 Fakultät für Informatik und Automatisierung      Fachgebiet: 2255

SWS nach Fachsemester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
										2	1	0									

### Lernergebnisse / Kompetenzen

Ziel des Kurses ist die Vermittlung von Wissen über die grundlegenden Aufgaben, Funktionsweisen und Eigenschaften von Kommunikationsmodellen. Die Kursteilnehmer lernen dabei verteilte Systeme als strukturierte Systeme aus Komponenten mit individuellen Aufgaben und komplexen Kommunikationsbeziehungen kennen, verstehen das Zusammenwirken dieser Komponenten und die grundsätzlichen Paradigmen, Modelle und Algorithmen, die dieses Zusammenwirken realisieren. Sie erwerben damit die Fähigkeit, problemspezifische Interaktionsmuster verteilter Systeme entwickeln und bezüglich ihrer Leistungen in unterschiedlichen Anwendungsdomänen zu analysieren und bewerten.

### Vorkenntnisse

Pflichtveranstaltung des Bachelor-Studiengangs Informatik der Semester 1-3

### Inhalt

Die Fähigkeit der Kommunikation ist eine der grundlegenden Eigenschaften verteilter IT-Systeme. Dieser Kurs vermittelt Grundlagenwissen über die zum Einsatz kommenden Kommunikationsmodelle in einem breiten Spektrum an Einsatzszenarien, beginnend bei eingebetteten verteilten Systemen bis hin zu globalen Informationssystemen. Ziel ist es, Wissen über die grundlegenden Aufgaben, Funktionsweisen und Eigenschaften von Kommunikationsmodellen zu vermitteln.

### Medienformen

Vorlesung mit Präsentation und Tafel, über Web-Plattform, Skript/Folien-Handouts, Bücher, Fachartikel; Übungsblätter, Diskussionsblätter

### Literatur

Siehe Webseiten der Veranstaltung

### Detailangaben zum Abschluss

Teil der Modulprüfung (60 min)

### verwendet in folgenden Studiengängen

Bachelor Informatik 2013

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik 2013 Vertiefung IN

Bachelor Informatik 2010





## Programmierparadigmen

Fachabschluss: über Komplexprüfung  
 Sprache: Deutsch

Art der Notengebung: unbenotet

Pflichtkennz.: Pflichtfach

Turnus: Sommersemester

Fachnummer: 5378

Prüfungsnummer: 2200377

Fachverantwortlich: Prof. Dr. Günter Schäfer

Leistungspunkte: 4      Workload (h): 120      Anteil Selbststudium (h): 75      SWS: 4.0  
 Fakultät für Informatik und Automatisierung      Fachgebiet: 2256

SWS nach Fachsemester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
										2	2	0									

### Lernergebnisse / Kompetenzen

Die Studierenden erwerben Basiswissen über Programmiersprachparadigmen, einschließlich der zugrunde liegenden Denk- und Verarbeitungsmodelle. Sie können Programmiersprachen und deren Konzepte nach wesentlichen Paradigmen klassifizieren (Fachkompetenz). Die Studierenden sind in der Lage, zu gegebenen Problemen geeignete Paradigmen kritisch auszuwählen. Sie können einfache Programme sowohl im funktionalen als auch im objektorientierten Programmierstil systematisch entwerfen und implementieren (Methodenkompetenz). Die Studierenden verstehen verschiedene Programmiersprachkonzepte im Kontext einer Programmiersprache (Systemkompetenz). Die Studierenden können erarbeitete Lösungen einfacher Programmieraufgaben in der Gruppe analysieren und bewerten (Sozialkompetenz).

### Vorkenntnisse

Algorithmen und Programmierung ( 1. Semester )

### Inhalt

Übersicht über behandelte Programmierparadigmen:

- Objektorientiertes Paradigma (Schwerpunkt)
- Funktionales Paradigma (Schwerpunkt)
- Nebenläufiges Paradigma (Schwerpunkt)
- Paralleles Paradigma
- Generisches Paradigma
- Aspektorientiertes Paradigma
- Generatives Paradigma

Demonstriert werden alle Schwerpunkt-Konzepte an der auf Java basierenden multi-paradigmatischen Sprache Scala.

### Medienformen

Präsentationen, Handouts

### Literatur

wird aktuell im Web veröffentlicht

### Detailangaben zum Abschluss

schriftliche Prüfung (60 min)

## verwendet in folgenden Studiengängen

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik 2013 Vertiefung IN

Bachelor Informatik 2010

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Elektrotechnik 2008 Vertiefung IN

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik 2008 Vertiefung IN

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Elektrotechnik 2013 Vertiefung IN

Bachelor Informatik 2013

## Modul: Softwaretechnik

Modulnummer 100334

Modulverantwortlich: Prof. Dr. Armin Zimmermann

Modulabschluss: Fachprüfung/Modulprüfung generiert

### Lernergebnisse

**Fachkompetenz:** Die Studierenden erwerben grundlegendes Wissen über Vorgehens- und Prozessmodelle der Softwareentwicklung, sowie über deren Methodik und Basiskonzepte. Sie können größere Entwicklungsaufgaben strukturieren, Lösungsmuster erkennen und anwenden, und verstehen den Entwurf von der Anforderungsermittlung bis hin zur Implementierung.

**Methodenkompetenz:** Den Studierenden wird Entscheidungskompetenz hinsichtlich möglicher Prinzipien, Methoden und Werkzeuge des ingenieurmäßigen Softwareentwurfs vermittelt.

**Systemkompetenz:** Die Studierenden verstehen das grundlegende Zusammenwirken unterschiedlicher Softwareentwicklungsphasen; anwendungsorientierte Kompetenzen bezüglich Modellierungsfähigkeit und Systemdenken werden geschult.

**Sozialkompetenz:** Die Studierenden verfügen über Fähigkeiten zur entwicklungsbezogenen, effektiven Teamarbeit.

### Voraussetzungen für die Teilnahme

Algorithmen und Programmierung

### Detailangaben zum Abschluss

## Softwaretechnik 1

 Fachabschluss: Prüfungsleistung schriftlich 90 min      Art der Notengebung: Gestufte Noten  
 Sprache: Deutsch      Pflichtkennz.: Pflichtfach      Turnus: Wintersemester

Fachnummer: 100533      Prüfungsnummer: 2200369

Fachverantwortlich: Prof. Dr. Armin Zimmermann

 Leistungspunkte: 3      Workload (h): 90      Anteil Selbststudium (h): 56      SWS: 3.0  
 Fakultät für Informatik und Automatisierung      Fachgebiet: 2236

SWS nach Fachsemester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
							2	1	0												

### Lernergebnisse / Kompetenzen

**Fachkompetenz:** Die Studierenden erwerben grundlegendes Wissen über Vorgehens- und Prozessmodelle der Softwareentwicklung, sowie über deren Methodik und Basiskonzepte. Sie können größere Entwicklungsaufgaben strukturieren, Lösungsmuster erkennen und anwenden, und verstehen den Entwurf von der Anforderungsermittlung bis hin zur Implementierung.

**Methodenkompetenz:** Den Studierenden wird Entscheidungskompetenz hinsichtlich möglicher Prinzipien, Methoden und Werkzeuge des ingenieurmäßigen Softwareentwurfs vermittelt.

**Systemkompetenz:** Die Studierenden verstehen das grundlegende Zusammenwirken unterschiedlicher Softwareentwicklungsphasen; anwendungsorientierte Kompetenzen bezüglich Modellierungsfähigkeit und Systemdenken werden geschult.

**Sozialkompetenz:** Die Studierenden verfügen über Fähigkeiten zur entwicklungsbezogenen, effektiven Teamarbeit.

### Vorkenntnisse

Algorithmen und Programmierung

### Inhalt

In der Lehrveranstaltung werden grundlegende Methoden, Modelle und Vorgehensweisen der Softwaretechnik bzw. des Software Engineering erlernt. Vorrangig wird die objektorientierte Sichtweise betrachtet, und in den Übungen anhand praktischer Beispiele vertieft. Für Implementierungsbeispiele wird vor allem JAVA verwendet.

- Einführung
- Modellierungskonzepte
  - . Überblick Modellierung
  - . klassische Konzepte (funktional, datenorientiert, algorithmisch, zustandsorientiert)
  - . Grundlagen Objektorientierung
  - . Unified Modeling Language (UML)
- Analyse
  - . Anforderungsermittlung
  - . Glossar, Geschäftsprozesse, Use Cases, Akteure
  - . Objektorientierte Analyse und Systemmodellierung
  - . Dokumentation von Anforderungen, Pflichtenheft
- Entwurf
  - . Software-Architekturen
  - . Objektorientiertes Design

- . Wiederverwendung (Design Patterns, Komponenten, Frameworks, Bibliotheken)
- Implementierung
  - . Konventionen und Werkzeuge
  - . Codegenerierung
  - . Testen
- Vorgehensmodelle
  - . Überblick, Wasserfall, Spiralmodell, V-Modell XT, RUP, XP
- Projektmanagement
  - . Projektplanung
  - . Projektdurchführung

## Medienformen

Vorlesungsfolien, auf den Webseiten verfügbar  
Übungsaufgaben, auf den Webseiten verfügbar

## Literatur

- Brügge, Dutoit: Objektorientierte Softwaretechnik. Pearson 2004
- Balzert: Lehrbuch der Software-Technik - Basiskonzepte und Requirements Engineering.
- sowie ergänzende Literatur, siehe Webseiten und Vorlesung

## Detailangaben zum Abschluss

### verwendet in folgenden Studiengängen

Master Mathematik und Wirtschaftsmathematik 2013 Vertiefung AM

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik 2013 Vertiefung IN

Bachelor Wirtschaftsinformatik 2013

Bachelor Ingenieurinformatik 2013

Bachelor Wirtschaftsinformatik 2015

Bachelor Informatik 2013

Master Mathematik und Wirtschaftsmathematik 2013 Vertiefung WM

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Elektrotechnik 2013 Vertiefung IN

## Softwaretechnik 2

Fachabschluss: Prüfungsleistung schriftlich 90 min Art der Notengebung: Gestufte Noten  
Sprache: deutsch Pflichtkennz.: Pflichtfach Turnus: Sommersemester

Fachnummer: 100564 Prüfungsnummer: 2200379

Fachverantwortlich: Prof. Dr. Detlef Streitferdt

Leistungspunkte: 3 Workload (h): 90 Anteil Selbststudium (h): 56 SWS: 3.0  
Fakultät für Informatik und Automatisierung Fachgebiet: 223A

SWS nach Fachsemester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
										2	1	0									

### Lernergebnisse / Kompetenzen

**Fachkompetenz:** Die Studierenden verfügen über anwendungsorientiertes Wissen zu Werkzeugen der Anforderungserhebung und –modellierung, der Prozessmodellierung und – anpassung, der Aufwandsschätzung, des Softwaretests, der Produktlinienentwicklung und der Wartung von Software.

**Methodenkompetenz:** Die Studierenden kennen den methodischen Hintergrund zu den vorgestellten Werkzeugen / Verfahren und sind daher in der Lage auch neue Problemstellungen zu lösen. Sie können aus den vorgestellten Methoden jeweils die passenden auswählen.

**Systemkompetenz:** Die Studierenden können die vorgestellten Methoden und Werkzeuge in Projekten unterschiedlicher Domänen anwenden.

**Sozialkompetenz:** Die Studierenden kennen die Bedeutung und den Einfluss der erlernten Methoden und Werkzeuge innerhalb einer Firma. Sie können daher Ihr jeweiliges Vorgehen und die Ergebnisse auf die Erfordernisse eines Projektes in einer Organisation abstimmen.

### Vorkenntnisse

Softwaretechnik 1

### Inhalt

Diese Vorlesung vertieft die Inhalte der Softwaretechnik. Durch den Anwendungsbezug und die vorgestellten Entwicklungswerkzeuge werden theoretische Kenntnisse umgesetzt. Die bekannten Phasen des Softwareentwicklungszyklus werden durch Themen vertieft, deren Bedeutung im industriellen Praxiseinsatz hoch ist.

- Requirements Engineering (RE) - Als eine der wichtigen Grundvoraussetzungen für hochwertige Systeme gilt die Requirements Engineering Phase. Die wichtigsten Technologien werden vorgestellt und eingesetzt.
  - Elicitation, Modeling, Validation/Verification
  - Goal-Oriented RE
  - Traceability
  - RE Tool Support
- Softwareprozessmodellierung - Nutzung und Anpassung von Entwicklungsprozessen mit zugehörigen Artefakten (z. B. Checklisten, Dokumentvorlagen, Werkzeugen, Rollenkonzept, ...). Je nach Anforderung, sollen einzelne oder ganze Prozesse erzeugt und effizient eingesetzt werden, um eine Entwicklergruppe bestmöglich zu unterstützen.

- Modellierung von Softwareentwicklungsprozessen (Wiederverwendung von Methoden- / Prozessschritten)
- Tailoring von SW-Entwicklungsprozessen

- Langlebige Systeme - Das Wissen um den Lebenszyklus von Softwaresystemen ist entscheidend für deren Entwicklung und zukünftigen Erfolg. Die geforderte Stabilität langlebiger Systeme (z. B. mehr als 30 Jahre) muss sich im Entwurf der Systeme wiederfinden.

- Design for Stability
- Reengineering
- Refactoring
- SW Wartung, Wartbarkeit

- Automatisiertes Testen - Veränderungen in den Anforderungen oder auch Fehlerbereinigungen führen zu der Notwendigkeit das System erneut testen zu müssen. Hierbei sind automatisierte Testansätze hilfreich. Zum einen lassen sie Änderungen an Testmodellen zu, aus denen Testfälle generiert werden. Zum anderen können Testfälle mit unterschiedlichen Zielen generiert werden, z. B. der Verbesserung der Codeabdeckung.

- Einordnung in den SW-Entwicklungsprozess
- Testmodellierung
- Testfallableitung
- Analyse von Testergebnissen

- Software Produktlinien - Der immer häufiger angewandte Produktlinienansatz erfordert ein Umdenken während des gesamten Entwicklungszyklus. Sollen später Produkte generiert und nicht jeweils als Eigenentwicklung entstehen, sind folgende Themen relevant:

- Merkmalmodelle (variable / gemeinsame Systemanteile)
- Produktlinien Architekturen
- Domänenspezifische Sprachen
- Testen von Produktlinien
- Generieren von Applikationen aus einer Produktlinie

## Medienformen

Bücher, Webseiten, Wissenschaftliche Paper, Open Source/Kommerzielle – Werkzeuge

## Literatur

[Boec 2004] Günter Böckle, Peter Knauber, Klaus Pohl, Klaus Schmid, "Software-Produktlinien: Methoden, Einführung und Praxis", dpunkt.Verlag GmbH, 2004.

[Broo 1995] Frederick P. Brooks, Jr., "The Mythical Man Month", Addison-Wesley, 1995.

[Fowl 1999] Martin Fowler, "Refactorings – Improving the Design of Existing Code", Addison Wesley, 1999.

[Gamm 1995] Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, "Design Pattern – Elements of Reusable Object-Oriented Software", Addison Wesley, 1995.

[Lams 2009] Axel van Lamsweerde, "Requirements Engineering: From System Goals to UML Models to Software Specifications", John Wiley & Sons, 2009.

[McCo 2006] Steve McConnell, "Software Estimation", Microsoft Press, 2006.

[Pohl 2005] Klaus Pohl, Günter Böckle, Frank van der Linden, "Software Product Line Engineering – Foundations, Principles, and Techniques", Springer, Heidelberg 2005.

[Pohl 2008] Klaus Pohl, "Requirements Engineering: Grundlagen, Prinzipien, Techniken", dpunkt.Verlag GmbH, 2008.

[Robe 1999] Suzanne Robertson, James Robertson, "Mastering the Requirements Process", Addison-Wesley, 1999.

[Rooc 2004] Stefan Roock, Martin Lippert, "Refactorings in großen Softwareprojekten", dpunkt.Verlag GmbH, 2004.

[Somm 2007] Ian Sommerville, "Software Engineering", Pearson Studium, 2007.

[Wieg 1999] Karl E. Wiegers, "Software Requirements", Microsoft Press, 1999.

[Your 1997] Edward Yourdon, "Death March", Prentice-Hall, 1997.



verwendet in folgenden Studiengängen

Bachelor Informatik 2013

Master Mathematik und Wirtschaftsmathematik 2013 Vertiefung AM

Master Mathematik und Wirtschaftsmathematik 2013 Vertiefung WM

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik 2013 Vertiefung IN

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Elektrotechnik 2013 Vertiefung IN

---

## Modul: Telematik 1

Modulnummer 100322

Modulverantwortlich: Prof. Dr. Günter Schäfer

Modulabschluss: Fachprüfung/Modulprüfung generiert

### Lernergebnisse

siehe Fachbeschreibung

### Voraussetzungen für die Teilnahme

### Detailangaben zum Abschluss

## Telematik 1

Fachabschluss: Prüfungsleistung schriftlich 90 min Art der Notengebung: Gestufte Noten  
 Sprache: Deutsch Pflichtkennz.: Pflichtfach Turnus: Sommersemester

Fachnummer: 100575 Prüfungsnummer: 2200383

Fachverantwortlich: Prof. Dr. Günter Schäfer

Leistungspunkte: 5 Workload (h): 150 Anteil Selbststudium (h): 105 SWS: 4.0  
 Fakultät für Informatik und Automatisierung Fachgebiet: 2253

SWS nach Fachsemester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
										3	1	0									

### Lernergebnisse / Kompetenzen

- **Fachkompetenz:** Die Studierenden verfügen über Kenntnisse und Überblickswissen zu Aufbau und Funktionsweise von Netzen, insbesondere des Internet.
- **Methodenkompetenz:** Die Studierenden sind in der Lage, einfache Protokollfunktionen zu spezifizieren und in Programmfragmente umzusetzen. Sie können die Auswirkungen bestimmter Entwurfsentscheidungen bei der Realisierung einzelner Protokollfunktionen auf grundlegende Leistungskenngrößen einschätzen. Sie kennen Darstellung von Protokollabläufen in Form von Message Sequence Charts und können gültige Protokollabläufe auf der Grundlage von Zustandsautomaten nachvollziehen.
- **Systemkompetenz:** Die Studierenden verstehen das grundsätzliche Zusammenwirken der Komponenten eines Netzes als System.
- **Sozialkompetenz:** Die Studierenden erarbeiten Problemlösungen einfacher Protokollfunktionen (z.B. Routing, Fehlerkontrolle, Flusskontrolle etc.) durch Bearbeiten von Übungsaufgaben in Gruppen und vertiefen bei Behandlung des Themas Geteilter Medienzugriff die technische Motivation für die Vorteile einer koordinierten Zusammenarbeit.
- Die Studierenden verfügen über Kenntnisse und Überblickswissen zu den anwendungsorientierten Schichten von Netzen und deren Protokolle, insbesondere des Internet. Die Studierenden kennen die grundlegenden Sicherheitsanforderungen an Kommunikationsdienste und Mechanismen zu ihrer Erfüllung.

### Vorkenntnisse

Hochschulzulassung;  
 Grundlagenvorlesung in Informatik oder Programmierung (z.B. „Algorithmen und Programmierung“ oder eine vergleichbare Grundlagenvorlesung)

### Inhalt

1. Einführung und Überblick: Grundsätzlicher Netzaufbau; Protokollfunktionen; Spezifikation; Architektur; Standardisierung; OSI- und Internet-Architekturmodell
2. Physikalische Schicht: Begriffe: Information, Daten und Signale; Physikalische Eigenschaften von Übertragungskanälen (Dämpfung, Verzerrung, Rauschen); Grenzen erreichbarer Datenübertragungsraten (Nyquist, Shannon); Taktsynchronisation; Modulationsverfahren (Amplituden-, Frequenz- und Phasenmodulation, kombinierte Verfahren)
3. Sicherungsschicht: Rahmensynchronisation; Fehlererkennung (Parität, Checksummen, Cyclic Redundancy Code); Fehlerbehebung (Forward Error Correction, Automatic Repeat Request); ARQ-Protokolle: Stop and Wait, Go-Back-N, Selective Reject; Medienzugriffsverfahren (ALOHA, Slotted ALOHA, Token-Ring, CSMA/CD); Ethernet; Internetworking: Repeater, Brücken und Router
4. Netzwerkschicht: Virtuelle Verbindungen vs. Datagramnetze; Aufgaben, Funktion und Aufbau eines Routers; Internet

Procol (IP): Paketaufbau und Protokollfunktionen, Hilfsprotokolle und Protokollversionen; Routingalgorithmen: Distanzvektor- und Link-State-Verfahren; Routingprotokolle des Internet (RIP, OSPF, BGP)

5. Transportschicht: Adressierung und Multiplexing; Verbindungsloser vs. verbindungsorientierter Transportdienst; Fehlerkontrolle; Flusskontrolle; Staukontrolle; Transportprotokolle des Internet (TCP, UDP)

6. Anwendungsorientierte Schichten: Sitzungsschicht, Darstellungsschicht und Anwendungsschicht, Grundarchitekturen verteilter Anwendungen: Client-Server, Peer-to-Peer, hybride Ansätze, Konkrete Protokolle der Anwendungsschicht: HTTP, SMTP, DNS;

7. Netzsicherheit

## Medienformen

Vorlesung mit Tafel und Folien-Präsentationen, Arbeitsblätter, Lehrbuch

## Literatur

A. S. Tanenbaum. Computernetzwerke. Pearson Education. · J. F. Kurose, K. W. Ross. Computernetze. Pearson Education.

## Detailangaben zum Abschluss

### verwendet in folgenden Studiengängen

Master Mathematik und Wirtschaftsmathematik 2013 Vertiefung AM

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik 2013 Vertiefung IN

Bachelor Ingenieurinformatik 2013

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Elektrotechnik 2013 Vertiefung IN

Bachelor Informatik 2013

Master Mathematik und Wirtschaftsmathematik 2013 Vertiefung WM

## Modul: Datenbank- und Betriebssysteme

Modulnummer 100335

Modulverantwortlich: Prof. Dr. Winfried Kühnhauser

Modulabschluss: Fachprüfung/Modulprüfung generiert

### Lernergebnisse

Thema in diesem Modul sind die grundlegenden Aufgaben, Paradigmen, funktionalen und nichtfunktionalen Eigenschaften zweier elementarer Themenkomplexe in der Informatik: den Datenbank- und Betriebssystemen. Es werden die elementaren Abstraktionen und Paradigmen vorgestellt und Kenntnisse über Prinzipien, Algorithmen und Datenstrukturen vermittelt, mit denen funktionale und nichtfunktionale Eigenschaften realisiert werden.

Die Teilnehmer lernen Datenbank- und Betriebssysteme als strukturierte parallele Systeme aus Komponenten mit individuellen Aufgaben und hochgradig komplexen Beziehungen verstehen. Sie erwerben die Fähigkeit, Systeme bezüglich ihrer Eignung und Leistungen in unterschiedlichen Anwendungsdomänen zu analysieren, zu bewerten und einzusetzen und kennen grundlegende theoretische Verfahren zu ihrem Entwurf und ihrer Implementierung.

### Voraussetzungen für die Teilnahme

### Detailangaben zum Abschluss

## Betriebssysteme

Fachabschluss: über Komplexprüfung  
 Sprache: Deutsch

Art der Notengebung: unbenotet

Pflichtkennz.: Pflichtfach

Turnus: Wintersemester

Fachnummer: 252

Prüfungsnummer: 2200322

Fachverantwortlich: Prof. Dr. Winfried Kühnhauser

Leistungspunkte: 4      Workload (h): 120      Anteil Selbststudium (h): 86      SWS: 3.0  
 Fakultät für Informatik und Automatisierung      Fachgebiet: 2255

SWS nach Fachsemester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
							2	1	0												

### Lernergebnisse / Kompetenzen

Die Kursteilnehmer sollen Betriebssysteme als strukturierte Systeme aus Komponenten mit individuellen Aufgaben und hochgradig komplexen Beziehungen verstehen; sie erwerben die Fähigkeit, Betriebssysteme bezüglich ihrer Eignung und Leistungen in unterschiedlichen Anwendungsdomänen zu analysieren, zu bewerten und einzusetzen.

### Vorkenntnisse

Rechnerorganisation, Rechnerarchitekturen 1, Programmierparadigmen, Kommunikationsmodelle, Algorithmen und Programmierung, Algorithmen und Datenstrukturen

### Inhalt

Betriebssysteme bilden das Software-Fundament aller informationstechnischen Systeme. Ihre funktionalen und vor allem ihre nichtfunktionalen Eigenschaften wie Robustheit, Sicherheit oder Wirtschaftlichkeit üben einen massiven Einfluss auf sämtliche Softwaresysteme aus, die unter ihrer Kontrolle ablaufen.

Dieser Kurs vermittelt Wissen über die grundlegenden Aufgaben, Funktionen und Eigenschaften von Betriebssystemen. Er stellt ihre elementaren Abstraktionen und Paradigmen vor und erklärt Prinzipien, Algorithmen und Datenstrukturen, mit denen funktionale und nichtfunktionale Eigenschaften realisiert werden.

### Medienformen

Vorlesung mit Projektor und Tafel, über Web-Plattform, Skript/Folien-Handouts, Bücher, Fachaufsätze, Übungsblätter, Diskussionsblätter

### Literatur

Siehe Webseiten der Veranstaltung

### Detailangaben zum Abschluss

schriftliche Modulprüfung

### verwendet in folgenden Studiengängen

Bachelor Ingenieurinformatik 2008

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik 2013 Vertiefung IN

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Elektrotechnik 2008 Vertiefung IN

Bachelor Mathematik 2013

Bachelor Wirtschaftsinformatik 2010

Bachelor Wirtschaftsinformatik 2011

Bachelor Informatik 2010

Master Mathematik und Wirtschaftsmathematik 2008

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Elektrotechnik 2013 Vertiefung IN

Bachelor Wirtschaftsinformatik 2009

Master Mathematik und Wirtschaftsmathematik 2013 Vertiefung AM

Bachelor Mathematik 2009

Bachelor Wirtschaftsinformatik 2013

Bachelor Ingenieurinformatik 2013

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik 2008 Vertiefung IN

Bachelor Wirtschaftsinformatik 2015

Bachelor Informatik 2013

Master Mathematik und Wirtschaftsmathematik 2013 Vertiefung WM

## Datenbanksysteme

Fachabschluss: über Komplexprüfung  
 Sprache: Deutsch

Art der Notengebung: unbenotet

Pflichtkennz.: Pflichtfach

Turnus: Wintersemester

Fachnummer: 244

Prüfungsnummer: 2200323

Fachverantwortlich: Prof. Dr. Kai-Uwe Sattler

Leistungspunkte: 4      Workload (h): 120      Anteil Selbststudium (h): 86      SWS: 3.0  
 Fakultät für Informatik und Automatisierung      Fachgebiet: 2254

SWS nach Fachsemester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
							2	1	0												

### Lernergebnisse / Kompetenzen

Nach dem Besuch dieser Veranstaltung können die Studierenden Datenbanksysteme anwenden. Sie kennen die Schritte des Entwurfs von Datenbanken und können die relationale Entwurfstheorie beschreiben. Weiterhin können sie deklarative Anfragen in SQL und XPath/XQuery formulieren sowie Integritätsbedingungen definieren.

Die Studierenden sind in der Lage, gegebene praktische Problemstellungen zu analysieren, im ER-Modell zu modellieren und in einer relationalen Datenbank abzubilden sowie SQL zur Anfrageformulierung zu nutzen.

### Vorkenntnisse

Vorlesung Algorithmen und Programmierung

### Inhalt

Grundbegriffe von Datenbanksystemen; Phasen des Datenbankentwurfs, Datenbankentwurf im Entity-Relationship-Modell, Relationaler Datenbankentwurf, Entwurfstheorie, Funktionale Abhängigkeiten und Normalformen; Grundlagen von Anfragen: Algebra und Kalküle; SQL: relationaler Kern und Erweiterungen, rekursive Anfragen mit SQL; Transaktionen und Integritätssicherung; Sichten und Zugriffskontrolle; XPath & XQuery als Anfragesprachen für XML

### Medienformen

Vorlesung mit Präsentation und Tafel, Handouts, Moodle

### Literatur

Saake, Sattler, Heuer: Datenbanken – Konzepte und Sprachen, 4. Auflage, mitp-Verlag, 2010.

### Detailangaben zum Abschluss

schriftliche Modulprüfung

### verwendet in folgenden Studiengängen

Bachelor Ingenieurinformatik 2008

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik 2013 Vertiefung IN

Bachelor Mathematik 2013



Bachelor Wirtschaftsinformatik 2010

Bachelor Wirtschaftsinformatik 2011

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Elektrotechnik 2013 Vertiefung IN

Bachelor Medientechnologie 2013

Bachelor Wirtschaftsinformatik 2009

Master Mathematik und Wirtschaftsmathematik 2013 Vertiefung AM

Bachelor Wirtschaftsinformatik 2013

Bachelor Ingenieurinformatik 2013

Bachelor Medientechnologie 2008

Bachelor Wirtschaftsinformatik 2015

Bachelor Informatik 2013

Master Mathematik und Wirtschaftsmathematik 2013 Vertiefung WM

## Modul: Algorithmen, Automaten und Komplexität

Modulnummer 100321

Modulverantwortlich: Prof. Dr. Dietrich Kuske

Modulabschluss: Fachprüfung/Modulprüfung generiert

### Lernergebnisse

Die Studierenden

- können grundlegende Algorithmen und Datenstrukturen anhand von Beispielen beschreiben,
- sind in der Lage, die Effizienz von Algorithmen zu bestimmen und
- können geeignete Algorithmen auswählen bzw. anpassen.

Die Studierenden

- kennen die Stufen 0,2 und 3 der Chomsky-Hierarchie, ihre automatentheoretischen Charakterisierungen und können die Umwandlungen ausführen,
- sind in der Lage, Nachweise der Nicht-Regularität zu führen,
- sind mit Abschluß- und algorithmischen Eigenschaften der regulären und kontextfreien Sprachen vertraut.

Die Studierenden

- kennen die Klassen der semi-entscheidbaren und der entscheidbaren Probleme mit wesentlichen Beispielen,
- sind in der Lage, die Church-Turing-These zu formulieren, ihre Bedeutung darzustellen und Argumente zu ihrer Begründung anzugeben.

Die Studierenden

- kennen die Komplexitätsklassen P und NP und einige NP-vollständige Probleme,
- können erläutern, warum die NP-schweren Probleme nicht effizient lösbar sind.

### Vorraussetzungen für die Teilnahme

### Detailangaben zum Abschluss

## Algorithmen und Datenstrukturen 1

 Fachabschluss: über Komplexprüfung  
 Sprache: Deutsch

Art der Notengebung: unbenotet

Pflichtkennz.: Pflichtfach

Turnus: Sommersemester

Fachnummer: 100534

Prüfungsnummer: 2200370

Fachverantwortlich: Prof. Dr. Martin Dietzfelbinger

Leistungspunkte: 3	Workload (h): 90	Anteil Selbststudium (h): 56	SWS: 3.0
Fakultät für Informatik und Automatisierung			Fachgebiet: 2242

SWS nach Fachsemester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
										2	1	0									

### Lernergebnisse / Kompetenzen

Die Studierenden kennen beispielhaft Methoden für die Spezifikation von Berechnungsproblemen und von abstrakten Datentypen und können sie für die grundlegenden Aufgaben und Datentypen anwenden. Sie können unter Verwendung der O-Notation die Laufzeit von (auch zusammengesetzten) iterativen und rekursiven Algorithmen analysieren. Sie können auf iterative und rekursive Algorithmen Induktionsargumente anwenden, um die Korrektheit zu zeigen. Die Studierenden kennen die grundlegenden Datenstrukturen „Array“, „Liste“, „Stack“, „Queue“, „gerichteter Baum“ und „Binärbaum“ mit ihren Implementierungsmöglichkeiten und können die zentralen Performanzparameter benennen und begründen. Sie kennen den Datentyp „binärer Suchbaum“ mit seinen Methoden für Einfügen und Suchen und den Datentyp „Mehrwegsuchbaum“. Die Studierenden kennen die Algorithmen für eine Variante von balancierten Binärbäumen und können sie an Beispielen durchführen. Die Studierenden kennen das Prinzip von einfachen Hashverfahren, verstehen die Funktionsweise und können das zu erwartende Verhalten für die verschiedenen Verfahren beschreiben. Sie kennen Konstruktionen einfacher randomisierter Hashklassen. Die Studierenden kennen die grundlegenden Sortieralgorithmen (Quicksort, Heapsort, Mergesort sowie Radixsort), können die Korrektheit der Verfahren begründen und ihre Laufzeit berechnen. Sie kennen den grundlegenden Datentyp „Priority Queue“ und seine Implementierung auf der Basis von binären Heaps. Die Studierenden kennen die Grundbegriffe der Graphentheorie und können mit ihnen umgehen. Sie kennen die wesentlichen Datenstrukturen für die Darstellung von Graphen und Digraphen mit den zugehörigen Methoden und Performanzparametern, sowie einfache Graphdurchmusterungsverfahren (Breitensuche, Tiefensuche). Sie kennen einen Algorithmus für die Ermittlung eines minimalen Spannbaums. Sie können die Korrektheit dieses Verfahrens begründen.

### Vorkenntnisse

Algorithmen und Programmierung, Mathematik I.

### Inhalt

Spezifikation von Berechnungsproblemen und von abstrakten Datentypen. Analyse von Algorithmen: Korrektheitsbeweise für iterative und rekursive Verfahren, Laufzeitbegriff, O-Notation, Laufzeitanalyse. Grundlegende Datenstrukturen (Listen, Stacks, Queues, Bäume). Binäre Suchbäume, Mehrwegsuchbäume, balancierte Suchbäume (AVL- und/oder Rot-Schwarz-Bäume, B-Bäume). Einfache Hashverfahren, universelles Hashing. Sortierverfahren: Quicksort, Heapsort, Mergesort, Radixsort. Priority Queues mit der Implementierung als Binärheaps. Grundbegriffe der Graphentheorie (ungerichtete und gerichtete Graphen, Markierungen an Knoten und Kanten, Wege und Kreise, Bäume und Wälder, Zusammenhangskomponenten). Datenstrukturen für Graphen (Adjazenzmatrix, Kantenliste, Adjazenzlisten, Adjazenzarrays). Durchmustern von Graphen: Breitensuche, Tiefensuche, Zusammenhangskomponenten, Entdecken von Kreisen. Minimale Spannbaume. Begleitend: Methoden für die Analyse von Laufzeit und Korrektheit.

### Medienformen

## Literatur

- \* G. Saake, K.-U. Sattler, Algorithmen und Datenstrukturen, 4. Auflage, dpunkt, 2010.
- \* R. Sedgewick, Algorithmen in C++, Pearson Studium, 2001.
- \* T. Ottmann, P. Widmayer, Algorithmen und Datenstrukturen, Spektrum Akademischer Verlag, 4., überarb. Aufl. 2002
- \* R. H. Güting, S. Dieker: Datenstrukturen und Algorithmen, Teubner, 2004
- \* T. Cormen, C. Leiserson, R. Rivest, C. Stein, Introduction to Algorithms, Second Edition, MIT Press 2001. (Auch auf deutsch erhältlich.)
- \* M. T. Goodrich, R. Tamassia, Data Structures and Algorithms in Java, 2. Auflage, Wiley, 2003.

## Detailangaben zum Abschluss

### verwendet in folgenden Studiengängen

Bachelor Ingenieurinformatik 2013

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Elektrotechnik 2013 Vertiefung IN

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik 2013 Vertiefung IN

## Automaten und Komplexität

Fachabschluss: über Komplexprüfung  
 Sprache: Deutsch

Art der Notengebung: unbenotet

Pflichtkennz.: Pflichtfach

Turnus: Sommersemester

Fachnummer: 100535

Prüfungsnummer: 2200371

Fachverantwortlich: Prof. Dr. Dietrich Kuske

Leistungspunkte: 3	Workload (h): 90	Anteil Selbststudium (h): 56	SWS: 3.0
Fakultät für Informatik und Automatisierung			Fachgebiet: 2241

SWS nach Fachsemester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
										2	1	0									

### Lernergebnisse / Kompetenzen

#### Die Studierenden

- kennen die Stufen 0, 2 und 3 der Chomsky-Hierarchie, ihre automatentheoretischen Charakterisierungen und können die Umwandlungen ausführen,
- sind in der Lage, Nachweise der Nicht-Regularität zu führen,
- sind mit Abschluss- und algorithmischen Eigenschaften der regulären und kontextfreien Sprachen vertraut.

#### Die Studierenden

- kennen die Klassen der semi-entscheidbaren und der entscheidbaren Probleme mit wesentlichen Beispielen,
- sind in der Lage, die Church-Turing These zu formulieren, ihre Bedeutung darzustellen und Argumente zu ihrer Begründung anzugeben.

#### Die Studierenden

- kennen die Komplexitätsklassen P und NP und einige NP-vollständige Probleme,
- können erläutern, warum die NP-schweren Probleme nicht effizient lösbar sind.

### Vorkenntnisse

Umgang mit mengentheoretischen Begriffen und Notationen (z.B. erworben in "Mathematik 1")

### Inhalt

- reguläre Sprachen: deterministische und nichtdeterministische endliche Automaten, reguläre Ausdrücke, effektiver Abschluss unter Booleschen Operationen, Verkettung und Iteration, Entscheidbarkeit von Leerheit, Inklusion und Äquivalenz, Pumping-Lemma
- kontextfreie Sprachen: Kellerautomaten, Parsing, effektiver Abschluss unter positiven Booleschen Operationen, Verkettung und Iteration, Entscheidbarkeit der Leerheit, Pumping-Lemma, Nichtabschluss unter Schnitt
- semi-entscheidbare Sprachen: Turing-Maschine
- Berechenbarkeit: Church-Turing These, Unentscheidbarkeit des Halteproblems und der Universalität von kontextfreien Sprachen
- Komplexitätstheorie: die Klassen P und NP, NP-Vollständigkeit, 3-SAT, graphentheoretische Probleme

### Medienformen

Tafel, Übungsblätter

## Literatur

Schöning "Theoretische Informatik kurzgefasst"

## Detailangaben zum Abschluss

## verwendet in folgenden Studiengängen

Bachelor Ingenieurinformatik 2013

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Elektrotechnik 2013 Vertiefung IN

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik 2013 Vertiefung IN

---

## Modul: Proseminar für IN Bsc

Modulnummer 100346

Modulverantwortlich: Prof. Dr. Winfried Kühnhauser

Modulabschluss:

Lernergebnisse

siehe Fachbeschreibung

Voraussetzungen für die Teilnahme

Detailangaben zum Abschluss

## Proseminar für IN Bsc

Fachabschluss: Studienleistung alternativ  
 Sprache: Deutsch

Art der Notengebung: Testat / Generierte Noten

Pflichtkennz.: Pflichtfach

Turnus: ganzjährig

Fachnummer: 100956

Prüfungsnummer: 2200410

Fachverantwortlich: Prof. Dr. Winfried Kühnhauser

Leistungspunkte: 2      Workload (h): 60      Anteil Selbststudium (h): 38      SWS: 2.0  
 Fakultät für Informatik und Automatisierung      Fachgebiet: 2255

SWS nach Fachsemester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
													0	2	0						

### Lernergebnisse / Kompetenzen

Studierende erarbeiten sich in kleinen Gruppen an praktischen Projektaufgaben fachliche Kompetenzen, praktisch-methodische Herangehensweisen und soziale Kompetenzen.

### Vorkenntnisse

fachspezifisch

### Inhalt

fachspezifisch

### Medienformen

fachspezifisch

### Literatur

fachspezifisch

### Detailangaben zum Abschluss

keine

### verwendet in folgenden Studiengängen

Bachelor Informatik 2013

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Elektrotechnik 2013 Vertiefung IN

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik 2013 Vertiefung IN



---

## Modul: Computergrafik

Modulnummer 100680

Modulverantwortlich: Prof. Dr. Beat Brüderlin

Modulabschluss: Fachprüfung/Modulprüfung generiert

### Lernergebnisse

siehe Fachbeschreibung

### Voraussetzungen für die Teilnahme

### Detailangaben zum Abschluss

## Computergrafik

 Fachabschluss: Prüfungsleistung schriftlich 60 min      Art der Notengebung: Gestufte Noten  
 Sprache: Deutsch      Pflichtkennz.: Pflichtfach      Turnus: Wintersemester

Fachnummer: 5367      Prüfungsnummer: 2200060

Fachverantwortlich: Prof. Dr. Beat Brüderlin

 Leistungspunkte: 5      Workload (h): 150      Anteil Selbststudium (h): 105      SWS: 4.0  
 Fakultät für Informatik und Automatisierung      Fachgebiet: 2252

SWS nach Fachsemester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
													3	1	0						

### Lernergebnisse / Kompetenzen

Vermitteln der Grundlagen der Computergrafik bestehend aus Lineare Algebra/homogene Vektorräumen, Physik des Lichts, Rasteroperationen, Bildsynthese, Bildverarbeitung und effiziente geometrische Algorithmen und Datenstrukturen. Die Vorlesung bildet die Grundlagen für "photorealistische" Bildsynthese, wie sie in der Industrie sowie bei den Medien Verwendung finden (z. B. Filmindustrie, Computer-Aided Design, Computerspiele, Styling). Vermittlung von Grundlagen für weiterführende Vorlesungen: Geometrisches Modellieren, Interaktive Grafische Systeme / Virtuelle Realität, Technisch-wissenschaftliche Visualisierung, Fortgeschrittene Bildsynthese, Bildverarbeitung I & II.

### Vorkenntnisse

Programmierkenntnisse Grundlagen Algorithmen &amp; Datenstrukturen

### Inhalt

Einführung: Überblick über das Fach Grafische Datenverarbeitung. Einführung: Vektoren und Matrizen, Transformationen, Homogene Vektorräume, 2D, 3D-Primitiven und Operationen, View-Transformationen Farbwahrnehmung, Tristimulus Ansatz, Farbmodelle: RGB, CMY, HSV, CIE. Spektrale Ansätze. Additive und Subtraktive Mischung. Lichtquellen und Filter. Rastergrafik-Hardware: Farbdiskretisierung, Farbbildröhre, LCD, Laserprinter, Ink-jet, etc. Rastergrafik: Rasterkonvertierung von Linien und Polygonen (Bresenham-Algorithmus, Polygonfüll-Algorithmus). Bildbearbeitung und Erkennung: Operationen auf dem Bildraaster, Bildtransformationen (Skalierung, Drehung), Resampling und Filterung (Bilinear, Gauß) Dithering, Antialiasing, Flood Filling, Kantenverstärkung (Kantenerkennung) Licht und Beleuchtung: (physikalische Größen: Wellenlänge, Leuchtdichte, Leuchtstärke), Wechselwirkung von Licht und Material, Lichtausbreitung und Reflexion, Refraktion, Beleuchtungsmodelle, Materialeigenschaften (geometrische Verteilung) Farbige Lichtquellen (spektrale Verteilung) (Phong: diffuse, spekulare Reflexion). Cook-Torrance, Mehrfachreflexion, Lichteffekte: Schatten, Halbschatten, Kaustik. Bildsynthese: Rendering basierend auf Rasterkonvertierung: Z-Buffer, Flat-Shading, Gouraud shading, Phong Shading Global Illumination, Raytracing, Photontracing, Radiosity Texturemapping / Image-based Rendering: Affines und perspektivisches Texturemapping, projektives Texturemapping, Environment Mapping, Bumpmaps Effiziente Datenstrukturen zum räumlichen Sortieren und Suchen. Kd-Tree, Hüllkörper-Hierarchie, Anwendungen in der Grafik Ray-tracing, Kollisionserkennung. OpenGL, GPU-Renderpipeline, Szenegraphen, Effizientes Rendering grosser Szenen. Ausblick: Überblick geometrischer und physikalischer Modelldatenstrukturen: CSG, B-Rep, Voxel, Octree, parametrische Flächen Computergrafische Animation: (Key frame, motion curve, physikalisch basiertes Modellieren, Kollisionserkennung, Molekülmodelle)

### Medienformen

Tafel, Folien, Buch Brüderlin, Meier: Computergrafik und geometrisches Modellieren (s. unten)

## Literatur

Brüderlin, B., Meier, A., Computergrafik und geometrisches Modellieren, Teubner-Verlag, 2001 Weiterführende Literatur: José Encarnaç o, Wolfgang Stra er, Reinhard Klein: Graphische Datenverarbeitung 1: Ger tetechnik, Programmierung und Anwendung graphischer Systeme. 4th, revised and extended edition, Oldenbourg, Munich, Germany, 1996. Jos  Encarnaç o, Wolfgang Stra er, Reinhard Klein: Graphische Datenverarbeitung 2: Modellierung komplexer Objekte und photorealistische Bilderzeugung. 4th, revised and extended edition, Oldenbourg, Munich, Germany, 1997. James D. Foley, Andries van Dam, Steven K. Feiner, John F. Hughes: Computer Graphics: Principles and Practice, Second Edition in C. - 2nd edition, Addison-Wesley, Reading, MA, USA, 1990. Alan Watt: 3D-Computergrafik. 3rd edition, Addison-Wesley, Reading, MA, USA, 2001.

## Detailangaben zum Abschluss

### verwendet in folgenden Studieng ngen

Bachelor Ingenieurinformatik 2008

Bachelor Polyvalenter Bachelor mit Lehramtsoption f r berufsbildende Schulen - Metalltechnik 2013 Vertiefung IN

Bachelor Polyvalenter Bachelor mit Lehramtsoption f r berufsbildende Schulen - Elektrotechnik 2008 Vertiefung IN

Bachelor Informatik 2013

Bachelor Mathematik 2013

Master Mathematik und Wirtschaftsmathematik 2013 Vertiefung WM

Bachelor Informatik 2010

Master Mathematik und Wirtschaftsmathematik 2008

Bachelor Polyvalenter Bachelor mit Lehramtsoption f r berufsbildende Schulen - Elektrotechnik 2013 Vertiefung IN

Master Mathematik und Wirtschaftsmathematik 2013 Vertiefung AM

Bachelor Mathematik 2009

Bachelor Polyvalenter Bachelor mit Lehramtsoption f r berufsbildende Schulen - Metalltechnik 2008 Vertiefung IN

---

## Modul: Logik und Logikprogrammierung

Modulnummer 100339

Modulverantwortlich: Prof. Dr. Dietrich Kuske

Modulabschluss: Fachprüfung/Modulprüfung generiert

### Lernergebnisse

Die Studierenden können zentrale Begriffe der Aussagen- und der Prädikatenlogik darstellen und auf konkrete Problemstellungen anwenden. Sie verstehen das Zusammenspiel von Syntax und Semantik formaler Systeme und argumentieren dabei präzise und folgerichtig. Sie kennen klassische Entscheidungsverfahren für die genannten Kalküle und können sie hinsichtlich ihrer Anwendbarkeit beurteilen. Sie haben ein grundlegendes Verständnis für die Logik-Programmierung und sind in der Lage, typische Probleme durch Prolog-Programme zu lösen.

### Voraussetzungen für die Teilnahme

### Detailangaben zum Abschluss

## Logik und Logikprogrammierung

Fachabschluss: Prüfungsleistung schriftlich 150 min Art der Notengebung: Gestufte Noten  
 Sprache: Deutsch Pflichtkennz.: Pflichtfach Turnus: Sommersemester

Fachnummer: 100574 Prüfungsnummer: 2200381

Fachverantwortlich: Prof. Dr. Dietrich Kuske

Leistungspunkte: 5 Workload (h): 150 Anteil Selbststudium (h): 94 SWS: 5.0  
 Fakultät für Informatik und Automatisierung Fachgebiet: 2241

	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
SWS nach Fachsemester	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
																3	2	0			

### Lernergebnisse / Kompetenzen

Die Studierenden können zentrale Begriffe der Aussagen- und der Prädikatenlogik darstellen und auf konkrete Problemstellungen anwenden. Sie verstehen das Zusammenspiel von Syntax und Semantik formaler Systeme und argumentieren dabei präzise und folgerichtig. Sie kennen klassische Entscheidungsverfahren für die genannten Kalküle und können sie hinsichtlich ihrer Anwendbarkeit beurteilen. Sie haben ein grundlegendes Verständnis für die Logik-Programmierung und sind in der Lage, typische Probleme durch Prolog-Programme zu lösen.

### Vorkenntnisse

sicherer Umgang mit mengentheoretischen Begriffen und Notationen (z.B. erworben in "Grundlagen und Diskrete Strukturen")

### Inhalt

Aussagenlogik: Syntax und Semantik, Erfüllbarkeit, Allgemeingültigkeit, Folgerung, Normalformen, Wahrheitstabelle, Resolution, natürliches Schließen, Tableau-Methode

Prädikatenlogik: Syntax und Semantik, Erfüllbarkeit, Allgemeingültigkeit, Folgerung, Normalformen, Resolution, Kompaktheitssatz, natürliches Schließen, Unentscheidbarkeit der Allgemeingültigkeit von Formeln der Prädikatenlogik und der Theorie der natürlichen Zahlen

PROLOG-Programmierung an typischen Problemklassen

### Medienformen

Folien, Übungsblätter, Vorlesungsmitschnitte (online)

### Literatur

Schöning "Logik für Informatiker"

Ivan Bratko: Prolog Programming for Artificial Intelligence. 4th edition, ISBN-10: 0321417461, ISBN-13: 9780321417466, Addison-Wesley, 2012.

### Detailangaben zum Abschluss

verwendet in folgenden Studiengängen

Bachelor Informatik 2013

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Elektrotechnik 2013 Vertiefung IN

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik 2013 Vertiefung IN

---

## **Modul: Wahlpflichtmodul(Wahl eines Moduls)**

Modulnummer101210

Modulverantwortlich: Prof. Dr. Wolfgang Fengler

Modulabschluss: Fachprüfung/Modulprüfung generiert

### **Lernergebnisse**

Die Modulbeschreibung ergibt sich aus den Beschreibungen der Module, die in diesem Modul enthalten sind.

### **Vorraussetzungen für die Teilnahme**

Die Modulbeschreibung ergibt sich aus den Beschreibungen der Module, die in diesem Modul enthalten sind.

### **Detailangaben zum Abschluss**

## Modul: Rechnerarchitekturen für IN

Modulnummer100331

Modulverantwortlich: Prof. Dr. Wolfgang Fengler

Modulabschluss: Fachprüfung/Modulprüfung generiert

### Lernergebnisse

**Fachkompetenz:** Die Studierenden verstehen detailliert Aufbau und Funktionsweise von Prozessoren, typischen Rechnerbaugruppen und deren Zusammenwirken. Sie verstehen detailliert Aufbau und Funktionsweise von fortgeschrittenen Prozessoren und Rechnern. Die Studierenden verstehen Entwicklungstendenzen der modernen Rechner- und Systemarchitektur.

**Methodenkompetenz:** Die Studierenden sind in der Lage, ein Beschreibungsmittel für die Modellierung von Strukturen und Abläufen mit formalen Mitteln anzuwenden. Die Studierenden entwerfen und analysieren einfache maschinennahe Programme. Die Studierenden konzipieren und entwerfen einfache Speicher- und E/A-Baugruppen. Die Studierenden sind in der Lage, Anwendungsbeispiele und Architekturvarianten zu entwickeln. Die Studierenden analysieren Leistungskennwerte von Rechnern und Rechnersystemen.

**Systemkompetenz:** Die Studierenden verstehen das Zusammenwirken der Funktionsgruppen von Rechnern als System und in Rechnersystemen. Sie erkennen den Zusammenhang zwischen Architektur und Anwendung auf dem Maschinenniveau anhand praktischer Übungen. Die Studierenden verstehen das Zusammenwirken der Funktionsgruppen von fortgeschrittenen Rechnern als System und in Rechnersystemen.

**Sozialkompetenz:** Die Studierenden sind in der Lage, praktische Problemstellungen der Rechnerarchitektur in der Gruppe zu lösen.

### Vorraussetzungen für die Teilnahme

Rechnerorganisation oder Technische Informatik

### Detailangaben zum Abschluss



## Rechnerarchitekturen 1

Fachabschluss: Prüfungsleistung schriftlich 90 min Art der Notengebung: Gestufte Noten  
 Sprache: Deutsch Pflichtkennz.: Pflichtfach Turnus: Sommersemester

Fachnummer: 5382 Prüfungsnummer: 2200265

Fachverantwortlich: Prof. Dr. Wolfgang Fengler

Leistungspunkte: 4 Workload (h): 120 Anteil Selbststudium (h): 75 SWS: 4.0  
 Fakultät für Informatik und Automatisierung Fachgebiet: 2231

SWS nach Fachsemester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
										2	2	0									

### Lernergebnisse / Kompetenzen

Fachkompetenz: Die Studierenden verstehen detailliert Aufbau und Funktionsweise von Prozessoren, typischen Rechnerbaugruppen und deren Zusammenwirken. Methodenkompetenz: Die Studierenden sind in der Lage, ein Beschreibungsmittel für die Modellierung von Strukturen und Abläufen mit formalen Mitteln anzuwenden. Die Studierenden entwerfen und analysieren einfache maschinennahe Programme. Die Studierenden konzipieren und entwerfen einfache Speicher- und E/A-Baugruppen. Systemkompetenz: Die Studierenden verstehen das Zusammenwirken der Funktionsgruppen von Rechnern als System und in Rechnersystemen. Sie erkennen den Zusammenhang zwischen Architektur und Anwendung auf dem Maschinenniveau anhand praktischer Übungen. Sozialkompetenz: Die Studierenden sind in der Lage, praktische Problemstellungen der Rechnerarchitektur in der Gruppe zu lösen.

### Vorkenntnisse

Vorlesung und Übung „Rechnerorganisation“

### Inhalt

Begriff der Rechnerarchitektur, Architekturmodellierung mit Petrinetzen, Innenarchitektur von Prozessoren, Befehlssatzarchitektur und Assemblerprogramme, Außenarchitektur von Prozessoren, Aufbau und Funktion von Speicherbaugruppen Aufbau und Funktion von Ein- und Ausgabebaugruppen, Zusammenwirken von Rechnerbaugruppen im Gesamtsystem

### Medienformen

Vorlesung: Folien (Beamer erforderlich), Arbeitsblätter (Online und Copyshop) Übung: Arbeitsblätter und Aufgabensammlung (Online und Copyshop) Selbststudium: Teleteaching-Kurs Allgemein: Webauftritt (Materialsammlung, Teleteaching-Kurs, Literaturhinweise, Links und weiterführende Infos) <http://tu-ilmenau.de/ra>

### Literatur

Primär: Eigenes Material - Materialsammlung zum Download - Materialsammlung im Copyshop - Teleteaching-Kurs  
 Sekundär: - W. Fengler, I. Philippow: Entwurf Industrieller Mikrocomputer-Systeme. ISBN 3-446-16150-3, Hanser 1991 (nur Kapitel 2). - C. Martin: Einführung in die Rechnerarchitektur - Prozessoren und Systeme. ISBN 3-446-22242-1, Hanser 2003. - T. Flik: Mikroprozessortechnik und Rechnerstrukturen. ISBN 3-540-22270-7, Springer 2005. Allgemein: Der primäre Anlaufpunkt ist der Webauftritt! <http://tu-ilmenau.de/ra> Dort gibt es die aktuellen Fassungen des Lehrmaterials sowie gelegentlich aktualisierte Literaturhinweise und Zusatzinfos.

### Detailangaben zum Abschluss

## verwendet in folgenden Studiengängen

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik 2013 Vertiefung IN

Bachelor Mathematik 2009

Bachelor Informatik 2010

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Elektrotechnik 2013 Vertiefung IN

Bachelor Informatik 2013

Bachelor Mathematik 2013

## Rechnerarchitekturen 2

Fachabschluss: Prüfungsleistung schriftlich 90 min Art der Notengebung: Gestufte Noten  
 Sprache: Deutsch Pflichtkennz.: Wahlpflichtfach Turnus: Wintersemester

Fachnummer: 5383 Prüfungsnummer: 2200055

Fachverantwortlich: Prof. Dr. Wolfgang Fengler

Leistungspunkte: 3 Workload (h): 90 Anteil Selbststudium (h): 56 SWS: 3.0  
 Fakultät für Informatik und Automatisierung Fachgebiet: 2231

SWS nach Fachsemester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
													2	1	0						

### Lernergebnisse / Kompetenzen

**Fachkompetenz:** Die Studierenden verstehen detailliert Aufbau und Funktionsweise von fortgeschrittenen Prozessoren und Rechnern. Die Studierenden verstehen Entwicklungstendenzen der modernen Rechner- und Systemarchitektur.  
**Methodenkompetenz:** Die Studierenden sind in der Lage, Anwendungsbeispiele und Architekturvarianten zu entwickeln. Die Studierenden analysieren Leistungskennwerte von Rechnern und Rechnersystemen.  
**Systemkompetenz:** Die Studierenden verstehen das Zusammenwirken der Funktionsgruppen von fortgeschrittenen Rechnern als System und in Rechnersystemen. Sie erkennen den Zusammenhang zwischen Architektur, Leistung und Anwendung anhand praktischer Übungen.  
**Sozialkompetenz:** Die Studierenden sind in der Lage, praktische Problemstellungen der Rechnerarchitektur in der Gruppe zu lösen.

### Vorkenntnisse

Vorlesung und Übung „Rechnerarchitekturen 1“ oder vergleichbare Veranstaltung

### Inhalt

Entwicklung der Prozessorarchitektur: Complex-Instruction-Set-Computing (CISC), Reduced-Instruction-Set-Computing (RISC); Befehls-Pipelining; Skalare Prozessorarchitektur, Very-Long-Instruction-Word-Architektur, Out of Order-Execution; Simultaneous Multithreading. Entwicklung der Speicherarchitektur: Adresspipelining, Burst Mode und Speicher-Banking; Speicherhierarchie, Cache-Prinzip, Cache-Varianten; Beispielarchitekturen; Spezialrechner: Aufbau eines Einchip-Controllers; Einchipmikrorechner des mittleren Leistungssegments, Erweiterungen im E/A-Bereich; Prinzip der digitalen Signalverarbeitung, Digitale Signalprozessoren (DSP), Spezielles Programmiermodell; Leistungsbewertung: MIPS, MFLOPS; Speicherbandbreite; Programmabhängiges Leistungsmodell (Benchmarkprogramme); Parallele Rechnerarchitekturen: Single Instruction Multiple Data, Multiple Instruction Single Data, Multiple Instruction Multiple Data; Enge und Lose Kopplung, Verbindungstopologien Entwicklung von Anwendungsbeispielen, Architekturvarianten und Berechnung von Leistungskennwerten

### Medienformen

Vorlesung: Folien (Beamer erforderlich), Arbeitsblätter (Online und Copyshop) Übung: Übungsmaterial (Online und Copyshop) Allgemein: Webseite (Materialsammlung und weiterführende Infos) <http://tu-ilmenau.de/ra>

### Literatur

Primär: Eigenes Material (Online und Copyshop) Sekundär: C. Märtin: Einführung in die Rechnerarchitektur - Prozessoren und Systeme. ISBN 3-446-22242-1, Hanser 2003. J. L. Hennessy, D. A. Patterson: Rechnerorganisation und -entwurf. ISBN 3-8274-1595-0, Elsevier 2005. W. Stallings: Computer Organization & Architecture. ISBN 0-13-035119-9, Prentice Hall 2003.

A. S. Tanenbaum, J. Goodman: Computerarchitektur. ISBN 3-8273-7016-7, Pearson Studium 2003. Allgemein: Der primäre Anlaufpunkt ist der Webaufttritt! <http://tu-ilmenau.de/ra> Dort gibt es die aktuellen Fassungen des Lehrmaterials sowie gelegentlich aktualisierte Literaturhinweise, Online-Quellen und Zusatzinfos.

## Detailangaben zum Abschluss

## verwendet in folgenden Studiengängen

Bachelor Ingenieurinformatik 2008

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik 2013 Vertiefung IN

Bachelor Mathematik 2009

Bachelor Informatik 2010

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Elektrotechnik 2013 Vertiefung IN

Bachelor Informatik 2013

Bachelor Mathematik 2013

## Praktikum Rechnerarchitekturen 1 und 2

Fachabschluss: Studienleistung alternativ  
 Sprache: deutsch

Art der Notengebung: Testat / Generierte Noten  
 Pflichtkennz.: Pflichtfach Turnus: ganzjährig

Fachnummer: 100561 Prüfungsnummer: 2200374

Fachverantwortlich: Dr. Bernd Däne

Leistungspunkte: 1 Workload (h): 30 Anteil Selbststudium (h): 19 SWS: 1.0  
 Fakultät für Informatik und Automatisierung Fachgebiet: 2231

SWS nach Fachsemester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
																0	0	1			

### Lernergebnisse / Kompetenzen

#### Fachkompetenz:

Die Studierenden verfügen über Kenntnisse und Überblickswissen zur Funktionsweise von Prozessoren und Rechnerstrukturen. Sie beherrschen den Umgang mit Beschreibungsmitteln und Modellen und erkennen das Zusammenwirken von Hardware und Software auf hardwarenahen Architekturebenen.

#### Methodenkompetenz:

Die Studierenden sind in der Lage, maschinennahe Programme zu verstehen, zu erstellen und in Betrieb zu nehmen. Sie sind in der Lage zur werkzeuggestützten Modellierung und zur Simulation und Analyse von Modellen.

#### Systemkompetenz:

Sie beherrschen den Umgang mit Werkzeugen zu Programmerstellung, Programmtest, Modellerstellung und Modellanalyse.

#### Sozialkompetenz:

Die Studierenden erarbeiten Problemlösungen gemeinsam in kleinen Gruppen.

### Vorkenntnisse

notwendig: Vorlesung und Übung Rechnerarchitekturen 1 (oder vergleichbare Veranstaltung)  
 empfohlen: Vorlesung und Übung Rechnerarchitekturen 2 (oder vergleichbare Veranstaltung)

### Inhalt

Einfache Assemblerprogramme  
 Ein- und Ausgabebaugruppen  
 Petri-Netze  
 Microcontroller  
 Fortgeschrittene Pipeline-Architekturen

### Medienformen

Laborpraktikum. Gedruckte Anleitungen, Hilfefunktionen der benutzten Software.

### Literatur

Aktuelle Literaturhinweise und weitere Quellen befinden sich stets auf den Internetseiten zur Lehrveranstaltung:  
<http://tu-ilmenau.de/?r-p-ra1>  
<http://tu-ilmenau.de/?r-p-ra2>

## Detailangaben zum Abschluss

## verwendet in folgenden Studiengängen

Bachelor Informatik 2013

Bachelor Mathematik 2013

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik 2013 Vertiefung IN

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Elektrotechnik 2013 Vertiefung IN

## Modul: Neuroinformatik und Schaltsysteme

Modulnummer 100332

Modulverantwortlich: Prof. Dr. Horst-Michael Groß

Modulabschluss: Fachprüfung/Modulprüfung generiert

### Lernergebnisse

**Fachkompetenz:** Die Studierenden lernen in diesem Modul die wesentlichen Grundlagen der sequentiellen und der parallelen (konnektionistischen) Informationsverarbeitung als die zwei wesentlichen Paradigmen der Informatik kennen.

Im Teil Neuroinformatik werden die Grundlagen der parallelen neurobiologischen Informationsverarbeitung und der darauf aufbauenden Neuroinformatik als wesentliche Säule der "Computational Intelligence" vermittelt. Die Studierenden verstehen die grundsätzliche Herangehensweise des konnektionistischen Ansatzes und kennen die wesentlichen biologischen Grundlagen, mathematischen Modellierungs- und algorithmischen Implementierungstechniken beim Einsatz von neuronalen und probabilistischen Methoden im Unterschied zu klassischen Methoden der Informations- und Wissensverarbeitung. Im Teil Schaltsysteme verfügen die Studierenden über Kenntnisse und vertieftes Wissen zu speziellen Strukturen und Funktionen von digitaler und programmierbarer Hardware und haben ein vertieftes Verständnis für die praktisch relevanten Problemstellungen und deren Komplexität.

**Methodenkompetenz:** Die Studierenden sind in der Lage, Fragestellungen aus den o. g. Problemkreisen zu analysieren, durch Anwendung des behandelten Methodenspektrums neue Lösungskonzepte zu entwerfen und umzusetzen, sowie bestehende Lösungen zu bewerten. Die Studierenden können ausgehend von einer Problemanalyse eigene Lösungen mit neuronalen Techniken erstellen. Sie sind in der Lage, komplexe digitale Schaltungen zu analysieren und zu synthetisieren und können auch umfangreichere Steuerungen sowohl mit Hilfe von diskreten Gatterschaltungen als auch mit Hilfe programmierbarer Schaltkreise erstellen. Sie können beim Entwurf systematisch vorgehen und ihre Entwürfe verifizieren.

**Systemkompetenz:** Auf Basis der vermittelten Methodik sind die Studierenden in der Lage, Methoden der Computational Intelligence auf neue Probleme anzuwenden und erfolgreich einzusetzen. Sie können dabei auf ein breites Methodenwissen aus den Bereichen der Neuroinformatik zurückgreifen.

Im Teil Schaltsysteme sind die Studierenden in der Lage, Programmsysteme zum Entwurf digitaler Steuerungen und Schaltungen anzuwenden.

**Sozialkompetenz:** Die Studierenden sind in der Lage, praktische Problemstellungen mit Methoden der parallelen und sequentiellen Informationsverarbeitung in der Gruppe zu analysieren, zu lösen und die Lösungen zu präsentieren. Sie erarbeiten Problemlösungen komplexer digitaler Schaltungen in der Gruppe, wobei einzelne Teilfunktionen von unterschiedlichen Personen entworfen werden. Sie können die von ihnen synthetisierten Schaltungen und Modellsteuerungen gemeinsam in einem Praktikum erproben, auf Fehler analysieren und korrigieren

### Vorraussetzungen für die Teilnahme

### Detailangaben zum Abschluss

## Neuroinformatik

Fachabschluss: über Komplexprüfung  
 Sprache: Deutsch

Art der Notengebung: unbenotet

Pflichtkennz.: Pflichtfach

Turnus: Sommersemester

Fachnummer: 1389

Prüfungsnummer: 2200343

Fachverantwortlich: Prof. Dr. Horst-Michael Groß

Leistungspunkte: 3	Workload (h): 90	Anteil Selbststudium (h): 68	SWS: 2.0
Fakultät für Informatik und Automatisierung			Fachgebiet: 2233

SWS nach Fachsemester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P

### Lernergebnisse / Kompetenzen

In der Vorlesung "Neuroinformatik" lernen die Studierenden die Grundlagen der Neuroinformatik und der Künstlichen Neuronalen Netze als wesentliche Säule der "Computational Intelligence" kennen. Sie verstehen die grundsätzliche Herangehensweise dieser Form der konnektionistischen Informations- und Wissensverarbeitung und kennen die wesentlichen Lösungsansätze, Modellierungs- und Implementierungstechniken beim Einsatz von neuronalen und probabilistischen Methoden im Unterschied zu klassischen Methoden der Informations- und Wissensverarbeitung. Die Studierenden sind in der Lage, Fragestellungen aus dem o. g. Problemkreisen zu analysieren, durch Anwendung des behandelten Methodenspektrums auf Fragestellungen aus den behandelten Bereichen (Mustererkennung, Signal- und Bildverarbeitung, Optimierung für Robotik, Control und Biomedizin) neue Lösungskonzepte zu entwerfen und umzusetzen sowie bestehende Lösungen zu bewerten.

### Vorkenntnisse

Keine

### Inhalt

Die Lehrveranstaltung vermittelt das erforderliche Methodenspektrum aus theoretischen Grundkenntnissen und praktischen Fähigkeiten zum Verständnis, zur Implementierung und zur Anwendung neuronaler und probabilistischer Techniken der Informations- und Wissensverarbeitung in massiv parallelen Systemen mit den Schwerpunkten Datenanalyse, Signalverarbeitung, Mustererkennung und Optimierung für verschiedene Ingenieursdisziplinen. Sie vermittelt sowohl Faktenwissen als auch begriffliches Wissen aus folgenden Themenbereichen:

- Informationsverarbeitung und Lernen in biologischen neuronalen Systemen
- Wichtige Neuronenmodelle (Biologisches Neuron, I&F Neuron, Formale Neuronen)
- Netzwerkmodelle - grundlegende Verschaltungsprinzipien & Architekturen
- Lernen in Neuronalen Netzen: wesentliche Arten des Lernens, wesentliche Lernparadigmen (Supervised / Unsupervised / Reinforcement Learning)
- Grundprinzip des überwachten Lernens: Multi-Layer-Perzeptron & Error-Backpropagation (EBP)-Lernregel
- Grundprinzip des unüberwachten Lernens: Self-Organizing Feature Maps (SOFM), Neural Gas, Growing Neural Gas ? als adaptive Vektorquantisierer
- Weitere wichtige Entwicklungen: Erweiterungen zum EBP-Algorithmus; Netzwerke mit Radialen Basisfunktionen, Support Vector Machines (SVM), Neuro-Fuzzy-Systeme, aktuelle Entwicklungen
- Anwendungsbeispiele aus den Bereichen Mustererkennung, Signal-/Bildverarbeitung, Biomedizin, Robotik, Neuro-Control
- exemplarische Software-Implementationen neuronaler Netze für nichtlineare Klassifikationsprobleme

Die Studierenden erwerben auch verfahrensorientiertes Wissen, indem für reale Klassifikations- und Approximationsprobleme verschiedene neuronale und probabilistische Lösungsansätze theoretisch behandelt und praktisch



umgesetzt werden. Dies ist auch Bestandteil des NI-Contests, der die softwaretechnische Implementierung eines Funktionsapproximators mittels eines überwacht trainierten Neuronales Netzes zum Gegenstand hat.  
SG BA-BMT: Im Rahmen des NI-Praktikums (0.5 SWS) werden die behandelten methodischen und technischen Grundlagen der neuronalen und probabilistischen Informationsverarbeitungs- und Lernprozesse durch die Studierenden mittels interaktiver Demo-Applets vertieft und in Gesprächsgruppen aufgearbeitet.

## Medienformen

Powerpoint-Folien, Demo-Applets, Videos

## Literatur

Zell, A.: Simulation Neuronaler Netzwerke. Addison-Wesley 1997  
Bishop, Ch.: Neural Networks for Pattern Recognition. Oxford Press, 1996  
Ritter, Martinetz, Schulten: Neuronale Netze. Addison-Wesley, Oldenbourg, 1994  
Görz, G., Rollinger, C.R., Schneeberger, J.: Handbuch der Künstlichen Intelligenz, Oldenbourg Verlag 2003  
Lämmel, Cleve: Künstliche Intelligenz – Lehr- und Übungsbuch. Fachbuchverlag, Leipzig, 2004

## Detailangaben zum Abschluss

### verwendet in folgenden Studiengängen

Master Wirtschaftsinformatik 2013  
Bachelor Ingenieurinformatik 2008  
Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik 2013 Vertiefung IN  
Master Wirtschaftsinformatik 2014  
Bachelor Biomedizinische Technik 2008  
Bachelor Informatik 2013  
Bachelor Mathematik 2013  
Bachelor Informatik 2010  
Master Mathematik und Wirtschaftsmathematik 2008  
Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Elektrotechnik 2013 Vertiefung IN  
Bachelor Mathematik 2009  
Bachelor Elektrotechnik und Informationstechnik 2008  
Bachelor Ingenieurinformatik 2013

## Schaltsysteme

Fachabschluss: über Komplexprüfung  
 Sprache: Deutsch

Art der Notengebung: unbenotet

Pflichtkennz.: Pflichtfach

Turnus: Sommersemester

Fachnummer: 100457

Prüfungsnummer: 2200344

Fachverantwortlich: Dr. Heinz-Dietrich Wuttke

Leistungspunkte: 3	Workload (h): 90	Anteil Selbststudium (h): 56	SWS: 3.0
Fakultät für Informatik und Automatisierung			Fachgebiet: 2235

SWS nach Fachsemester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P

### Lernergebnisse / Kompetenzen

#### Lernziele:

- vertiefende Vermittlung von fundierten Kenntnissen und Fertigkeiten zum Entwurf digitaler Systeme,
- Einbeziehung verallgemeinerter Wertverlaufsgleichheiten,
- Herausbildung von Fähigkeiten zur kritischen Beurteilung von entworfenen Schaltsystemen bzgl. Aufwand und Korrektheit sowie zur praktischen Fehlersuche in Hard- und Softwarerealisierungen

#### Fachkompetenz:

Die Studierenden verfügen über Kenntnisse und vertieftes Wissen zu speziellen Strukturen und Funktionen von digitaler und programmierbarer Hardware und haben ein vertieftes Verständnis für die praktisch relevanten Problemstellungen und deren Komplexität.

#### Methodenkompetenz:

Die Studierenden sind in der Lage, komplexe digitale Schaltungen zu analysieren und zu synthetisieren. Sie können auch umfangreichere Steuerungen sowohl mit Hilfe von diskreten Gatterschaltungen als auch mit Hilfe programmierbarer Schaltkreise erstellen. Sie können beim Entwurf systematisch vorgehen und ihre Entwürfe verifizieren.

#### Systemkompetenz:

Die Studierenden sind in der Lage, Programmsysteme zum Entwurf digitaler Steuerungen und Schaltungen anzuwenden.

#### Sozialkompetenz:

Die Studierenden erarbeiten Problemlösungen komplexer digitaler Schaltungen in der Gruppe, wobei einzelne Teilfunktionen von unterschiedlichen Personen entworfen werden. Sie können die von ihnen synthetisierten Schaltungen und Modellsteuerungen gemeinsam in einem Praktikum erproben, auf Fehler analysieren und korrigieren

### Vorkenntnisse

- erfolgreicher Abschluß des Fachs "Rechnerorganisation"
- Grundkenntnisse im Entwurf kombinatorischer und sequentieller Schaltungen

### Inhalt

#### Einführung

## **Entwurf kombinatorischer Schaltungen**

- Verallgemeinerte Wertverlaufsgleichheit
- Implizite Gleichungssysteme
- Struktursynthese, Minimierung
- Dynamische Probleme

## **Entwurf sequentieller Automaten**

- Partielle, nichtdeterminierte Automaten
- Struktursynthese mit unterschiedlichen Flip-Flop-Typen
- Operations- und Steuerwerke

## **Entwurf paralleler Automaten**

- Komposition/ Dekomposition
- Automatenetze

### Entwurfswerkzeuge

## Medienformen

Vorlesung mit Tafel und PowerPoint, Video zur Vorlesung, Applets und PowerPoint-Präsentationen im Internet, Arbeitsblätter, Lehrbuch

## Literatur

- Wuttke, Henke: Schaltsysteme, Pearson-Verlag, München 2003
- -Informatik-Duden: Duden-Verlag 1988/89 Schiffmann,
- S. Hentschke: Grundzüge der Digitaltechnik, Teubner-Verlag, Stuttgart 1988
- T. Flick, H. Liebig: Mikroprozessortechnik, 4. Auflage, Springer- Verlag, Berlin 1994
  
- Literaturempfehlungen zu den Vorlesungen

## Detailangaben zum Abschluss

## verwendet in folgenden Studiengängen

Master Wirtschaftsinformatik 2013

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik 2013 Vertiefung IN

Master Wirtschaftsinformatik 2014

Bachelor Ingenieurinformatik 2013

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Elektrotechnik 2013 Vertiefung IN

Bachelor Informatik 2013

Bachelor Mathematik 2013

## Praktikum Neuroinformatik und Schaltsysteme

Fachabschluss: Studienleistung

Art der Notengebung: Testat / Generierte Noten

Sprache: Deutsch

Pflichtkennz.: Pflichtfach

Turnus: Sommersemester

Fachnummer: 100536

Prüfungsnummer: 2200372

Fachverantwortlich: Dr. Klaus Debes

Leistungspunkte: 2	Workload (h): 60	Anteil Selbststudium (h): 49	SWS: 2.0
Fakultät für Informatik und Automatisierung			Fachgebiet: 2233

SWS nach Fachsemester	1.FS			2.FS			3.FS			4.FS			5.FS			6.FS			7.FS		
	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P	V	S	P
													0	0	2						

### Lernergebnisse / Kompetenzen

Fachkompetenz:

SS: Durch die vertiefende Vermittlung von fundierten Kenntnissen und Fertigkeiten zum Entwurf digitaler, sequentieller Steuerungssysteme sowie Möglichkeiten zu deren formaler Beschreibung und Verifikation.

NI: Durch die Vermittlung von fundierten Grundlagen zur Funktionsweise von überwachten und unüberwachten neuronalen Lernverfahren.

Methodenkompetenz:

SS: Die Studierenden sind in der Lage, digitale Steuerungen zu analysieren, zu optimieren und zu synthetisieren. Sie können digitale Steuerungen sowohl mit Hilfe von diskreten Gatterschaltungen als auch mit Hilfe programmierbarer Schaltkreise erstellen. Fähigkeiten zur kritischen Beurteilung von entworfenen Schaltsystemen bzgl. Aufwand und Korrektheit befähigen zur praktischen Fehlersuche in Hard- und Softwarerealisierungen.

NI: Die Studierenden sind in der Lage, die Funktionsweise neuronaler Lernverfahren zu verstehen, zu analysieren und eigenständig für unterschiedliche Problemstellungen zu entwerfen.

Systemkompetenz:

SS: Mit Hilfe formaler Methoden können sie digitale Steuerungssysteme analysieren und validieren.

NI: Erlernen formaler Methoden zur Anwendung neuronaler überwachter und unüberwachter Lernverfahren;

Sozialkompetenz:

SS: Die Studierenden erarbeiten Problemlösungen einfacher digitaler Schaltungen in der Gruppe. Sie können die von ihnen synthetisierten Schaltungen gemeinsam in einem Praktikum auf Fehler analysieren und korrigieren.

NI: Die Studierenden analysieren auf der Basis vorgegebener Applets in kleinen Gruppen und stellen ihre Ergebnisse vor.

### Vorkenntnisse

Vorlesung Neuroinformatik, Vorlesung Schaltsysteme

### Inhalt

NI: Die behandelten methodischen und technischen Grundlagen der neuronalen und probabilistischen Informationsverarbeitungs- und Lernprozesse werden mittels interaktiver Demo-Applets vertieft und in Gesprächsgruppen aufgearbeitet; dazu Erarbeitung eines Protokolls im Selbststudium mit Überprüfung der Ergebnisse in einer Präsenzveranstaltung (Arbeitsaufwand 15 Zeitstunden)

SS: Durchführung von zwei Versuchen

- Hardware-Realisierung sequentieller Schaltungen
- PLD-Realisierung sequentieller Schaltungen

## Medienformen

Versuchsanleitungen, Internetpräsenz, Applets

## Literatur

NI: Praktikumsanleitung und Vorlesungsunterlagen

SS:

- H.D. Wuttke, K. Henke: Schaltsysteme – Eine Automatenorientierte Einführung – Pearson Education, 2006
- H.D. Wuttke, K. Henke: Schaltsysteme, Arbeitsblätter, Übungsaufgaben, Praktikumsanleitung, TU Ilmenau, [www.tu-ilmenau.de/iks](http://www.tu-ilmenau.de/iks)
- W. Schiffmann, H. Schmitz: Technische Informatik, Band I und II, Springer-Verlag, 2004
- V. Claus, A. Schwill: Informatik-Duden, Bibliographisches Institut, 2006
- Spezielle Literatur in den Versuchsanleitungen und unter [www.tu-ilmenau.de/iks](http://www.tu-ilmenau.de/iks)

## Detailangaben zum Abschluss

## verwendet in folgenden Studiengängen

Bachelor Informatik 2013

Bachelor Mathematik 2013

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Metalltechnik 2013 Vertiefung IN

Bachelor Ingenieurinformatik 2013

Bachelor Polyvalenter Bachelor mit Lehramtsoption für berufsbildende Schulen - Elektrotechnik 2013 Vertiefung IN



## **Glossar und Abkürzungsverzeichnis:**

LP	Leistungspunkte
SWS	Semesterwochenstunden
FS	Fachsemester
V S P	Angabe verteilt auf Vorlesungen, Seminare, Praktika
N.N.	Nomen nominandum, Nomen nescio, Platzhalter für eine noch unbekannte Person (wikipedia)
Objekttypen lt. Inhaltsverzeichnis	K=Kompetenzfeld; M=Modul; P,L,U= Fach (Prüfung,Lehrveranstaltung,Unit)