

## Modulbeschreibung

# IN0009: Grundlagen: Betriebssysteme und Systemsoftware

Fakultät für Informatik

<b>Modulniveau:</b> Bachelor	<b>Sprache:</b> Deutsch	<b>Semesterdauer:</b> Einsemestrig	<b>Häufigkeit:</b> Wintersemester
<b>Credits*:</b> 6	<b>Gesamtstunden:</b> 180	<b>Eigenstudiumsstunden:</b> 105	<b>Präsenzstunden:</b> 75

\* Die Zahl der Credits kann in Einzelfällen studiengangsspezifisch variieren. Es gilt der im Transcript of Records oder Leistungsnachweis ausgewiesene Wert.

### Beschreibung der Studien-/Prüfungsleistungen:

Die Modulprüfung besteht aus einer Klausur von 90 Minuten, in der die Studierenden das Verständnis des Stoffes (wie die Verwaltung von Betriebsmitteln und der Einsatz von Systemsoftware) durch Wiedergabe und Anwendung des Gelernten nachweisen müssen. Zudem müssen sie vorgestellte Verfahren anwenden und Ihre Fähigkeiten in der Systemprogrammierung bei der Lösung kleiner Aufgaben unter Beweis stellen.

### Wiederholungsmöglichkeit:

Im Folgesemester: Nein

Am Semesterende: Ja

### (Empfohlene) Voraussetzungen:

IN0001 Einführung in die Informatik 1 und IN0004 Einführung in die Rechnerarchitektur werden empfohlen

### Inhalt:

Grundkonzepte: Betriebssysteme; Nebenläufigkeit; Parallele Programmierung; Systemnahe Programmierung (Prozesse, Speicher, Kommunikation, BM-Verwaltung; Modelle (abstrakt, formal) für Nebenläufigkeit, u.a. Petrinetze; Wechselseitiger Ausschluss, Synchronisation, Deadlocks; Compiler/Linker/Loader mit Integration von Bibliotheken, Übergang auf (geeignete) Hardware-Basis, maschinennahe Programmierung und C; I/O insbesondere zur Vorbereitung der Vernetzung)

### Lernergebnisse:

Nach Abschluss des Moduls sind die Studierenden in die Lage, die Grundlagen, Probleme und Lösungen von Betriebssystemen und der Systemsoftware zu verstehen und die aktuellen Entwicklungen einzuschätzen. Zudem verstehen sie die einzelnen Komponenten, wie zum Beispiel die Prozess- und die Speicherverwaltung, und sie können die unterschiedlichen Strategien und Verfahren analysieren und bewerten. Sie sind zudem in die Lage, die erworbenen Grundlagenkenntnisse direkt auf neue Entwicklungen im Bereich der Betriebssysteme sowie der Systemsoftware anzuwenden, indem sie Betriebssystemkomponenten und andere Systemsoftware selber entwickeln.

### Lehr- und Lernmethoden:

Die Vorlesung führt mittels einer Folienpräsentation und anhand von Beispielen in die Grundprinzipien und Vorgehensweisen des Betriebssystemsbaus und deren interner Funktionalität ein.

Begleitende Übungen vertiefen das Verständnis der in der Vorlesung vermittelten Konzepte anhand von geeigneten Gruppenaufgaben und veranschaulichen die Anwendung unterschiedlicher Herangehensweisen anhand von

überschaubaren Aufgabenstellungen der unterschiedlichen Aspekte der Betriebssystemunterteilung. Zusätzliche Programmieraufgaben ermöglichen es den Studierenden ihr Wissen im Selbststudium zu vertiefen. Rückmeldung und Hilfe in Programmierübungen helfen zusätzlich den Lernfortschritt zu beurteilen und ihr Können zu verbessern.

**Medienform:**

Unterlagen (Folien und weitere Dokumente) via moodle

**Literatur:**

A.S. Tanenbaum, H. Bos: Modern Operating Systems, 4/E (Pearson, 2015)

**Modulverantwortliche(r):**

Baumgarten, Uwe; Prof. Dr. rer. nat. habil.: [baumgaru@tum.de](mailto:baumgaru@tum.de)

**Lehrveranstaltungen (Lehrform, SWS) Dozent(in):**

0000000052 Übungen zu Grundlagen: Betriebssysteme und Systemsoftware (IN0009) (2SWS UE, WS 2020/21) [BF]  
Ott J [L], Uhl M

0000004057 Übungen zu Grundlagen: Betriebssysteme und Systemsoftware (IN0009) (2SWS UE, WS 2020/21) [BF]  
Ott J [L], Uhl M

240984991 Grundlagen: Betriebssysteme und Systemsoftware (IN0009) (3SWS VO, WS 2020/21) [BF]  
Ott J [L], Ott J, Uhl M

---

Weitere Informationen zum Modul und seiner Zuordnung zum Curriculum:

<https://campus.tum.de/tumonline/wbModHb.wbShowMHBReadOnly?pKnotenNr=452822>

Generiert am: 22.01.2021 22:56

## Modulbeschreibung

# IN0010: Grundlagen: Rechnernetze und Verteilte Systeme

Fakultät für Informatik

<b>Modulniveau:</b> Bachelor	<b>Sprache:</b> Deutsch	<b>Semesterdauer:</b> Einsemestrig	<b>Häufigkeit:</b> Sommersemester
<b>Credits*:</b> 6	<b>Gesamt- stunden:</b> 180	<b>Eigenstudiums- stunden:</b> 105	<b>Präsenz- stunden:</b> 75

\* Die Zahl der Credits kann in Einzelfällen studiengangsspezifisch variieren. Es gilt der im Transcript of Records oder Leistungsnachweis ausgewiesene Wert.

### Beschreibung der Studien-/Prüfungsleistungen:

Die Prüfungsleistung wird in Form einer 90-minütigen Klausur erbracht. Verständnisfragen sowie Rechenaufgaben überprüfen die Vertrautheit der in der Vorlesung behandelten Technologien und Methoden von Rechnernetzen und Verteilten Systemen sowie das durch Implementierung von Protokollmechanismen gewonnene Verständnis. Rechenaufgaben überprüfen darüber hinaus die Fähigkeit, die Leistungsfähigkeit ausgewählter Netze und verteilter Anwendungen zu bestimmen.

### Wiederholungsmöglichkeit:

Im Folgesemester: Nein  
Am Semesterende: Ja

### (Empfohlene) Voraussetzungen:

IN0001 Einführung in die Informatik 1, IN0003 Einführung in die Informatik 2 bzw. ab WiSe 2018/19 Funktionale Programmierung und Verifikation, IN0002 Praktikum: Grundlagen der Programmierung

### Inhalt:

- Rechnernetze
- ++ Überblick: Computernetze und das Internet
- +++ Komponenten (Router, Switches, Clients, Server)
- +++ Aufbau (Topologien, Routing, Pakete gegenüber virtuelle Verbindungen)
- +++ Schichtenmodell (OSI und Internet)
- +++ Geschichtlicher Hintergrund
- ++ Schichtenübergreifende Konzepte (innerhalb der Schichten behandelt):
- +++ Adressierung
- +++ Fehlererkennung
- +++ Codierung und Modulation
- +++ Medienzugriffsverfahren
- +++ Flusskontrolle
- +++ Verbindungsmanagement
- +++ Paketvermittlung gegenüber virtuelle Verbindungen
- ++ Schichten:
- +++ Anwendungsprotokolle und Anwendungen (application layer)

- ++++ Aufgaben und Interface
- ++++ Beispiele: HTTP, DNS, SMTP (Mail), Peer-to-Peer-Protokolle
- +++ Transportschicht
- ++++ Aufgabe und Interface
- ++++ Beispiele: TCP und UDP
- +++ Vermittlungsschicht
- ++++ Aufgaben und Interface
- ++++ Routing: Link State gegenüber Distance Vector Protokolle
- ++++ Adressierung: IP Adressen
- ++++ Beispiele: IP, Routing im Internet
- +++ Sicherungsschicht
- ++++ Aufgaben und Interface
- ++++ Beispiele: Ethernet, Wireless LAN
- +++ Bitübertragungsschicht
- ++++ Aufgaben und Interface
- ++++ Beispiele
- Verteilte Systeme:
- ++ Middleware, z.B. RPC
- ++ Web Services
- Übergreifende Aufgaben:
- ++ Netzmanagement
- ++ IT-Sicherheit
- +++ Kryptographische Mechanismen und Dienste
- +++ Authentifizierung, Vertraulichkeit, Integrität
- +++ Protokolle mit Sicherheitsmechanismen, z.B.: IPsec, PGP, Kerberos, SSL, SSH, ...
- +++ Firewalls, Intrusion Detection

#### Inhalt der Übung:

Die Übung behandelt Rechenaufgaben zur Bestimmung der Leistungsfähigkeit von Protokollen und Mechanismen einzelner Schichten (Physical Layer, Data Link Layer, Network Layer, Transport Layer).

In Programmierübungen wird die Implementierung einzelner Protokollmechanismen geübt.

#### Lernergebnisse:

Nach dem Modul "Grundlagen: Rechnernetze und Verteilte Systeme" sind die Studierenden in der Lage, die Technologien und Methoden von Rechnernetzen und Verteilten Systemen zu verstehen, und anhand wesentlicher Protokolle der geschichteten Netzarchitektur zu erläutern, welche Protokollmechanismen in den einzelnen Schichten eingesetzt werden, und wie diese funktionieren. Zudem verstehen sie, wie verteilte Anwendungen wie z.B. das World Wide Web mit Hilfe der Internetprotokolle realisiert sind, und wie Rechnernetze aufgebaut sind. Darüber hinaus sind die Studierenden in der Lage, die Leistungsfähigkeit ausgewählter Netze und verteilter Anwendungen zu bestimmen, sowie einzelne Protokollmechanismen zu implementieren.

#### Lehr- und Lernmethoden:

Die interaktive Vorlesung mit Folienpräsentationen, Animationen, Vorführungen und Live-Programmierung stellt das Grundwissen zu Rechnernetzen und Verteilten Systemen vor und erläutert sie an Beispielen. Quizze helfen den Studierenden zu erkennen, ob sie die Grundbegriffe und wesentliche Zusammenhänge verstanden haben. Hausaufgaben ermöglichen Studierenden die Themen im Selbststudium zu vertiefen. Begleitende Tutorübungen vertiefen anhand geeigneter Aufgaben das Verständnis der Inhalte der Vorlesung und zeigen die Anwendung der verschiedenen Methoden anhand überschaubarer Problemstellungen. Die Präsentation der eigenen Lösung in der begleitenden Tutorübung verbessert die Kommunikationsfähigkeiten und erlaubt, den eigenen Lernfortschritt mit dem anderer Studierender zu vergleichen. Programmieraufgaben erlauben rechnergestützter Vertiefung sowie Anwendung konzeptionellen Wissens auf praktische Problemstellungen.

#### Medienform:

Präsentationsfolien, Übungsblätter, Beispieldemonstrationen

#### Literatur:

Literaturangaben sind auf den Webseiten der Veranstaltung und auf den Vorlesungsfolien angegeben.

Standardwerke sind u.a.:

1. James F. Kurose, Keith W. Ross

Computernetzwerke

Pearson Studium; 5. aktualisierte Auflage, 2012

2. Andrew S. Tanenbaum / Prof. David J. Wetherall

Computernetzwerke

Pearson Studium, 5. aktualisierte Auflage, 2012

**Modulverantwortliche(r):**

Carle, Georg; Prof. Dr.-Ing.: [carle@tum.de](mailto:carle@tum.de)

**Lehrveranstaltungen (Lehrform, SWS) Dozent(in):**

000000708 Grundlagen: Rechnernetze und Verteilte Systeme (IN0010) (3SWS VO, SS 2020/21) [GP]

Carle G [L], Carle G, Günther S, Stubbe H

0000001126 Übungen zu Grundlagen: Rechnernetze und Verteilte Systeme (IN0010), Mo, Di (2SWS UE, SS 2020/21) [GP]

Carle G [L], Günther S, Stubbe H, Andre J, Helm M, Jelten J

0000021126 Übungen zu Grundlagen: Rechnernetze und Verteilte Systeme (IN0010), Mi, Do, Fr (2SWS UE, SS 2020/21) [GP]

Carle G [L], Günther S, Stubbe H, Andre J, Jelten J

---

Weitere Informationen zum Modul und seiner Zuordnung zum Curriculum:

<https://campus.tum.de/tumonline/wbModHb.wbShowMHBReadOnly?pKnotenNr=458318>

Generiert am: 22.01.2021 22:57

## Modulbeschreibung

# ED0421: Praktikum Software- und Datenbanktechnik für Berufliche Bildung

Fakultät TUM School of Education

<b>Modulniveau:</b> Master	<b>Sprache:</b> Deutsch/Englisch	<b>Semesterdauer:</b> Zweimestrig	<b>Häufigkeit:</b> Wintersemester/Sommersemester
<b>Credits*:</b> 9	<b>Gesamtstunden:</b> 270	<b>Eigenstudiumsstunden:</b> 210	<b>Präsenzstunden:</b> 60

\* Die Zahl der Credits kann in Einzelfällen studiengangsspezifisch variieren. Es gilt der im Transcript of Records oder Leistungsnachweis ausgewiesene Wert.

### Beschreibung der Studien-/Prüfungsleistungen:

Die Modulprüfung besteht aus dem Entwurf, der Realisierung und der Präsentation eines größeren Softwareproduktes mit Anbindung an eine selbstentworfenen relationale Datenbank. Die Studierenden müssen dabei nachweisen, dass sie über grundlegende theoretische Kenntnisse und Fertigkeiten aus der Softwaretechnik ebenso wie über Datenbanksysteme verfügen. Ein wichtiger Bestandteil der Projektarbeit ist auch die Dokumentation der Projektarbeit in Form einer Ausarbeitung im Umfang von mindestens 20 Seiten je Studierende/r, die auch die Bezüge zur Theorie klarstellen muss. Die abschließende Präsentation im Umfang von mindestens 20 Minuten je Studierende/r muss über Projektplanung, -durchführung und -ergebnisse berichten und diese mit Hilfe theoretischer Konzepte begründen. Weiter muss das Produkt vorgestellt und seine Kosten (d.h. der Arbeitsaufwand) dargestellt werden. An den Vortrag schließt sich eine Fragerunde im Umfang von mindestens 10 Minuten je Studierende/r an, in der die Studierenden vor allem ausreichende theoretische Kenntnisse aus den Bereichen Software- und Datenbanktechnik nachweisen müssen.

### Wiederholungsmöglichkeit:

Im Folgesemester: Nein

Am Semesterende: Ja

### (Empfohlene) Voraussetzungen:

Module IN0001, IN0002, IN0006

### Inhalt:

Anwendung der grundlegenden Konzepte und Fertigkeiten aus der Softwaretechnik, insbesondere der klassischen Phasen der Softwareentwicklung (Anforderungsanalyse, Entwurf; Implementierung; Test; Installation und Wartung), Pflichten- und Lastenheft, Klassische und moderne Vorgehensmodelle, Modellierungstechniken für Analyse und Entwurf, Architektur- und Entwurfsmuster, grundlegende Techniken des Projektmanagements sowie Risiken und typische Probleme in Softwareprojekten.

Erwerb und Anwendung von grundlegenden Kenntnissen und Fertigkeiten zu Datenbanksystemen, insbesondere zur Datenmodellierung, zur relationalen Entwurfstheorie, Optimierung auf der Grundlage der Relationenalgebra, zur Abfragesprache SQL, zu Datenintegrität, zur relationalen physischen Datenorganisation (Speicherorganisation, Indexstrukturen), Anfragebearbeitung, Transaktionsverwaltung, Grundzügen der Fehlerbehandlung (Recovery, Backup) und der Mehrbenutzersynchronisation sowie zu Sicherheitsaspekten wie Autorisierung.

**Lernergebnisse:**

Die Teilnehmer sind in der Lage, ein kleines Softwaresystem im Team zu konzipieren, zu implementieren und zu testen sowie dessen Ergebnisse fachgerecht zu dokumentieren und zu präsentieren. Die Studierenden verfügen über grundlegende Kenntnisse und Fertigkeiten zu informatischen und ingenieurwissenschaftlichen Konzepten, Methoden und Modellen der Software- und Datenbanktechnik. Sie beherrschen die systematische Nutzung eines Datenbanksystems vom konzeptuellen Entwurf über den Implementationsentwurf zum physischen Entwurf. Sie können auch komplexe Anfragen in SQL formulieren und haben ein Grundverständnis der logischen und physischen Optimierung auf der Grundlage der Relationenalgebra. Weiterhin haben sie den sicheren Betrieb hinsichtlich Recovery, Mehrbenutzersynchronisation und Autorisierung verstanden. Sie kennen die Risiken und die typischen Probleme in Softwareprojekten und können den Aufwand solcher Projekte mit Hilfe spezieller Methoden (z.B. Cocomo) abschätzen. Sie sind auch in der Lage, die Eignung und den zeitlichen Aufwand für den Einsatz ähnlicher Projekte im Informatikunterricht abzuschätzen. In der Projektarbeit zeigen die Studierenden insbesondere, dass sie in der Lage sind, all diese Kenntnisse und Fertigkeiten im Kontext eines größeren Vorhabens anzuwenden, zu verbinden und auch zu kommunizieren.

**Lehr- und Lernmethoden:**

Den theoretischen Hintergrund zur Softwaretechnik sollten die Studierenden bereits in ihrem Bachelorstudium erworben haben. Die Kenntnisse und Fertigkeiten zu Datenbanken erwerben sie im Rahmen einer Vorlesung oder im angeleiteten Selbststudium auf der Grundlage geeigneter Literatur oder mit Hilfe von Online-Kursen. Die eigentliche Projektarbeit wird durch Präsenzveranstaltungen begleitet und strukturiert. In diesen Veranstaltungen werden die Vorgaben, aber auch die Ergebnisse der Arbeitsphasen diskutiert und besprochen. Die Teilnehmer üben systematisches Software-Engineering dabei in Kleingruppen nach Vorgabe und mit Zeitkontrolle (Entwurf, Implementierung, Test). Die Teilergebnisse der Gruppenarbeit sind in Vorträgen zu präsentieren. Entwurf, Projektplanung und Implementierung sind zu dokumentieren. Das Projekt wird auch im Hinblick auf seine Eignung für den Informatikunterricht beurteilt und bewertet.

**Medienform:**

Vorlesung mit animierten Folien, Webschnittstelle für SQL, Online-Materialien und -Kurse, Präsentationsfolien, Entwicklungs- und Simulationssoftware. Studierende bekommen auf Wunsch nach individueller Absprache mit den Dozenten weitere Hilfestellungen wie zusätzliche Literatur und/oder Online-Lerntools genannt.

**Literatur:**

Ludewig, J., Lichter, H. 2010. Software Engineering. Grundlagen, Menschen, Prozesse, Techniken. dpunkt-Verl, Heidelberg.  
Hubwieser, P., Löffler, P., Schwaiger, P., Spohrer, M., Steinert, M., Voß, S., Winhard, F. 2009. Rekursive Datenstrukturen, Softwaretechnik. Schülerbuch - Jahrgangsstufe 11. Informatik 4. Klett, Stuttgart.  
Kemper A., Eickler A. 2015. Datenbanksysteme. Eine Einführung. 10., aktualisierte und erweiterte Auflage, Oldenbourg Verlag.  
Kemper A., Wimmer M. 2012. Übungsbuch: Datenbanksysteme. 3. Auflage Oldenbourg Verlag.  
Silberschatz A., Korth H. F., Sudarshan S. 2010. Database System Concepts. Sixth Edition, McGraw-Hill.

**Modulverantwortliche(r):**

keine Angabe: [keine Angabe](#)

**Lehrveranstaltungen (Lehrform, SWS) Dozent(in):**

000001483 Softwareentwicklungsprojekt (für Lehramtskandidaten) (6SWS PR, WS 2020/21) [BF]  
Engel M

240952278 Grundlagen: Datenbanken (IN0008) (3SWS VO, WS 2020/21) [BF]  
Kemper A, Anneser C, Schmeißer J, Sichert M, Vogel L

---

Weitere Informationen zum Modul und seiner Zuordnung zum Curriculum:

<https://campus.tum.de/tumonline/wbModHb.wbShowMHBReadOnly?pKnotenNr=2651631>

Generiert am: 22.01.2021 22:58

## Modulbeschreibung

# ED0315: Theoretische Informatik für Berufliche Bildung

Fakultät TUM School of Education

---

<b>Modulniveau:</b> Master	<b>Sprache:</b> Deutsch	<b>Semesterdauer:</b> Einsemestrig	<b>Häufigkeit:</b> Sommersemester
<b>Credits*:</b> 4	<b>Gesamt- stunden:</b> 120	<b>Eigenstudiums- stunden:</b> 60	<b>Präsenz- stunden:</b> 60

\* Die Zahl der Credits kann in Einzelfällen studiengangsspezifisch variieren. Es gilt der im Transcript of Records oder Leistungsnachweis ausgewiesene Wert.

### Beschreibung der Studien-/Prüfungsleistungen:

Die Modulprüfung ist eine Klausur von 120 Minuten. Die gestellten Fragen decken die Lernergebnisse der Lehrveranstaltung ab und orientieren sich an den in den Übungsanteilen bearbeiteten Aufgaben. Insbesondere soll die Klausur auf das Staatsexamen vorbereiten und orientiert sich deshalb an dort geforderten Aufgaben. Dabei wird Wert auf eine Mischung von anwendungsbezogenen und deklarativen Aufgaben gelegt.

### Wiederholungsmöglichkeit:

Im Folgesemester: Nein

Am Semesterende: Ja

### (Empfohlene) Voraussetzungen:

ED0338

### Inhalt:

Es werden folgende grundlegende Konzepte der Theoretischen Informatik behandelt: Formale Sprachen und ihre Einordnung in die Chomsky-Hierarchie. Elementare Automatentheorie und der Bezug zu formalen Sprachen. Der algorithmische Begriff der Entscheidbarkeit. Grundlagen der Komplexitätstheorie. Grundlagen der Berechenbarkeit

### Lernergebnisse:

Nach der erfolgreichen Teilnahme an diesem Modul verstehen die Studierenden die wesentlichen Konzepte der Theoretischen Informatik auf einem grundlegenden, aber wissenschaftlichen Niveau. Die Studierenden können reguläre Ausdrücke, kontextfreie Grammatiken, die Chomsky-Hierarchie, endliche Automaten und Turingmaschinen auf entsprechende Aufgabenstellungen anwenden. Sie können die grundlegenden Konzepte der Komplexitätstheorie erklären. Sie können die Unterschiede zwischen LOOP, WHILE und GOTO Berechenbarkeit erklären.

### Lehr- und Lernmethoden:

In der Vorlesung wechseln sich Vorträge, Präsentationen und Übungsphasen in Einzel- und Gruppenarbeit ab. Das sofortige Einüben des Gelernten an geeigneten Aufgaben wird besonders betont.

### Medienform:

Folienpräsentation, Tafelanschrieb, Übungsblätter



**Literatur:**

John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman: Einführung in Automatentheorie, Formale Sprachen und Berechenbarkeit

Uwe Schöning: Theoretische Informatik kurzgefasst

**Modulverantwortliche(r):**

keine Angabe: [keine Angabe](#)

**Lehrveranstaltungen (Lehrform, SWS) Dozent(in):**

Vorlesung/Übung:

Theoretische Informatik für Berufliche Bildung (4 SWS)

n.n.

---

Weitere Informationen zum Modul und seiner Zuordnung zum Curriculum:

<https://campus.tum.de/tumonline/wbModHb.wbShowMHBRReadOnly?pKnotenNr=1301959>

Generiert am: 22.01.2021 23:00

## Modulbeschreibung

### IN2209: IT Sicherheit

#### Fakultät für Informatik

---

<b>Modulniveau:</b> Bachelor/Master	<b>Sprache:</b> Deutsch	<b>Semesterdauer:</b> Einsemestrig	<b>Häufigkeit:</b> Wintersemester
<b>Credits*:</b> 7	<b>Gesamt- stunden:</b> 210	<b>Eigenstudiums- stunden:</b> 135	<b>Präsenz- stunden:</b> 75

\* Die Zahl der Credits kann in Einzelfällen studiengangsspezifisch variieren. Es gilt der im Transcript of Records oder Leistungsnachweis ausgewiesene Wert.

#### Beschreibung der Studien-/Prüfungsleistungen:

Die Prüfungsleistung wird in Form einer 105-minütigen Klausur erbracht. Wissensfragen überprüfen die Vertrautheit mit Konzepten und Vorgehensweisen zur Absicherung von Komponenten und Systemen. Mit kleineren Aufgaben wird die Fähigkeit überprüft, sichere Protokolle zu entwickeln, aber auch Sicherheitskonzepte gezielt und korrekt einzusetzen, oder aber Sicherheitsmängel zu erkennen.

#### Wiederholungsmöglichkeit:

Im Folgesemester: Nein

Am Semesterende: Ja

#### (Empfohlene) Voraussetzungen:

IN0009 Grundlagen: Betriebssysteme und Systemsoftware, IN0010 Grundlagen: Rechnernetze und Verteilte Systeme, Grundkenntnisse in der Kryptographie sind hilfreich

#### Inhalt:

Grundlegende Begriffe und Sicherheitsprobleme, Kryptographische Grundlagen, Schlüsselaustausch und -management, Digitale Identität, Netzwerksicherheit, Anwendungssicherheit, Systemsicherheit, Sicherheitsanalyse und Security Engineering

#### Lernergebnisse:

Nach der erfolgreichen Teilnahme an diesem Modul verstehen die Teilnehmer die wesentlichen Konzepte, Methoden und Mechanismen zum Schutz von Daten und Systemen vor Manipulation und Missbrauch auf einem grundlegenden, praxis-orientierten, wissenschaftlichen Niveau. Sie sind dann in der Lage, die Konzepte zur Erhöhung der Sicherheit korrekt einzusetzen, Sicherheitsprotokolle zu entwickeln und diese zu bewerten. Sie verstehen die Ursachen von Sicherheits-Problemen heutiger Systeme und beherrschen die wichtigsten Konzepte zu deren Absicherung. Sie sind deshalb in der Lage, eigenständig mögliche Sicherheitschwachstellen in Designs oder Protokollen zu identifizieren und auf der Grundlage der erlernten Methoden und Konzepte Lösungen zu entwickeln.

#### Lehr- und Lernmethoden:

Vorlesung ergänzt durch Übung, um konzeptionelle und technische Sachverhalte anhand konkreter Aufgabenstellungen, einschließlich praktischer Programmieraufgaben, detailliert zu durchdringen und gezielt zu erweitern.

**Medienform:**

Vorlesungsfolien

**Literatur:**

- IT-Sicherheit: Konzepte, Verfahren, Protokolle, Claudia Eckert, 10. Auflage, De Gruyter, 2018.
- Understanding Cryptography, C. Paar und J. Pelzl, 2. Auflage Springer, 2010.
- Introduction to Computer Security, Matt Bishop, Addison-Wesley, 2004.
- Applied Cryptography, Bruce Schneier, 2. Auflage, John Wiley & Sons, 1996.
- Computer Security: Principles and Practice, Global Edition. W. Stallings, L. Brown, Pearson, 4th edition, 2018
- Exploiting Software: How to break code, G. Hogl und und G. McGraw, Addison-Wesley, 2004.

**Modulverantwortliche(r):**

Eckert, Claudia; Prof. Dr.: [claudia.eckert@tum.de](mailto:claudia.eckert@tum.de)

**Lehrveranstaltungen (Lehrform, SWS) Dozent(in):**

0000002721 Übung zu IT Sicherheit (IN2209) (1SWS UE, WS 2020/21) [BF]

Eckert C, Franzen F

240919971 IT Sicherheit (IN2209) (4SWS VO, WS 2020/21) [BF]

Eckert C, Franzen F, Peuckert L

---

Weitere Informationen zum Modul und seiner Zuordnung zum Curriculum:

<https://campus.tum.de/tumonline/wbModHb.wbShowMHBReadOnly?pKnotenNr=1697995>

Generiert am: 22.01.2021 23:03

## Modulbeschreibung

### ED0287: Didaktik der Informatik

Fakultät TUM School of Education

<b>Modulniveau:</b> Master	<b>Sprache:</b> Deutsch	<b>Semesterdauer:</b> Zweisemestrig	<b>Häufigkeit:</b> Wintersemester
<b>Credits*:</b> 6	<b>Gesamt- stunden:</b> 180	<b>Eigenstudiums- stunden:</b> 120	<b>Präsenz- stunden:</b> 60

\* Die Zahl der Credits kann in Einzelfällen studiengangsspezifisch variieren. Es gilt der im Transcript of Records oder Leistungsnachweis ausgewiesene Wert.

#### Beschreibung der Studien-/Prüfungsleistungen:

Die Prüfungsleistung ist ein Lernportfolio, das sich über beide Lehrveranstaltungen erstreckt. Das Lernportfolio soll insbesondere den Lernfortschritt in Bezug auf die Erscheinungsformen der informatischen Bildung aufzeigen. Dazu wird die genaue Analyse von curricularen Vorgaben und Standards mit dem Planen und Evaluieren von Unterrichtssituationen verbunden

#### Wiederholungsmöglichkeit:

Im Folgesemester: Ja  
Am Semesterende: Nein

#### (Empfohlene) Voraussetzungen:

Grundlagen der praktischen Informatik,  
Mathematische und strukturelle Grundlagen,  
Praktische Informatik,  
Korrektheit und funktionale Programmierung

#### Inhalt:

- Grundlagen informatikbezogenen Lehrens und Lernens  
Ziele, Erscheinungs- und Organisationsformen informatischer Bildung  
grundlegende Methoden der Informatik (z. B. Beweisen, Modellieren, Strukturieren, Konstruieren, Validieren)  
Techniken und Hilfsmittel zur Vermittlung informatischer Inhalte (z. B. Programmiersprachen, Modellierungssprachen, Entwicklungsumgebungen; Hardwareumgebungen; Simulationssysteme)  
Analyse und didaktische Aufbereitung geeigneter Praxisfelder  
Besondere pädagogische Zielsetzung des Informatikunterrichts: Datenschutz, informationelle Selbstbestimmung, Gefahren bei der Nutzung von Rechnernetzen.
- Konzeption, Gestaltung und Bewertung von Informatikunterricht  
Historische und aktuelle Unterrichtsansätze  
Typische Unterrichtsmethoden der Informatik  
Didaktische (Re-)Konstruktion informatischen Wissens, insbesondere von didaktische Reduktion  
Planung, Organisation und Durchführung von Informatikunterricht  
Analyse und Bewertung von Lehr- und Lernprozessen im Informatikunterricht  
Vertiefung der Grundlagen informatikbezogenen Lehrens und Lernens  
Erweiterung der Kenntnisse und Fertigkeiten zur Konzeption, Gestaltung und Bewertung von Informatikunterricht.

**Lernergebnisse:**

Die Teilnehmer kennen die wichtigsten Ziele, Erscheinungs- und Organisationsformen informatischer Bildung; sie beherrschen die wichtigsten Methoden zur Planung und Durchführung von Informatikunterricht. Sie können Lehr-Lernprozesse und Unterrichtsansätze analysieren und hinsichtlich ihrer Wirksamkeit bewerten.

**Lehr- und Lernmethoden:**

Die unterschiedlichen Inhalte werden mit verschiedenen den Lerninhalten angepassten Lehr/Lernmethoden aufbereitet. Dabei bilden Präsentationen und Kurzreferate das theoretische Grundgerüst. In den Arbeitsphasen der Vorlesung, und insbesondere in der Übung werden Gruppen- und Einzelarbeiten durchgeführt, um die theoretischen Grundlagen intensiv selbst zu erfahren.

**Medienform:**

E-Learning System Moodle, Folien, Quellen

**Literatur:**

Modul IN2152: Hubwieser P.: Didaktik der Informatik. Grundlagen, Konzepte und Beispiele. Springer Verlag, Berlin, 3. Auflage, 2007.

**Modulverantwortliche(r):**

keine Angabe: [keine Angabe](#)

**Lehrveranstaltungen (Lehrform, SWS) Dozent(in):**

0000003761 Didaktik der Informatik 1 für Berufliche Bildung (2SWS VO, WS 2020/21) [GP]

821091997 Didaktik der Informatik 1 (2SWS VO, WS 2020/21) [BF]  
Hubwieser P

0000001519 Didaktik der Informatik 2 für Berufliche Bildung (2SWS VO, SS 2020/21) [GP]

0000001538 Didaktik der Informatik - Didaktik der Informatik 2 (2SWS VO, SS 2020/21) [GP]  
Geldreich K, Hubwieser P

---

Weitere Informationen zum Modul und seiner Zuordnung zum Curriculum:

<https://campus.tum.de/tumonline/wbModHb.wbShowMHBRReadOnly?pKnotenNr=1018853>

Generiert am: 22.01.2021 23:04

## Modulbeschreibung

# ED0211: Hauptseminar Didaktik der Informatik mit Schulpraktikum

Fakultät TUM School of Education

<b>Modulniveau:</b> Master	<b>Sprache:</b> Deutsch	<b>Semesterdauer:</b> Einsemestrig	<b>Häufigkeit:</b> Wintersemester/Sommersemester
<b>Credits*:</b> 6	<b>Gesamtstunden:</b> 180	<b>Eigenstudiumsstunden:</b> 90	<b>Präsenzstunden:</b> 90

\* Die Zahl der Credits kann in Einzelfällen studiengangsspezifisch variieren. Es gilt der im Transcript of Records oder Leistungsnachweis ausgewiesene Wert.

### Beschreibung der Studien-/Prüfungsleistungen:

Die Prüfungsleistung besteht aus einer schriftlichen wissenschaftlichen Ausarbeitung, die aus zwei Teilen besteht. Der Umfang beträgt 10-20 Seiten. Der erste Teil der Arbeit hat streng wissenschaftlichen Charakter, der zweite Teil hat Berichtscharakter. Zur Entzerrung der Prüfbelastung im Sinne der Studierenden kann die Prüfung beider Teile getrennt voneinander durchgeführt werden. Die Ausarbeitung wird durch einen 15-30-minütigen Vortrag ergänzt. Dabei zeigen die Studierenden, dass sie in der Lage sind, ein Thema der Fachdidaktik Informatik aus wissenschaftlicher Sicht bzw. im Kontext eigener Unterrichtserfahrungen zu betrachten. Die Themen generieren sich aus der aktuellen fachdidaktischen Forschung. Eine Bearbeitung in Kleingruppen ist möglich, wobei der Berichtsteil in Einzelarbeit zu erbringen ist.

### Wiederholungsmöglichkeit:

Im Folgesemester: Ja

Am Semesterende: Nein

### (Empfohlene) Voraussetzungen:

IN0001: Einführung in die Informatik 1

IN0003: Einführung in die Informatik 2

IN0006: Einführung in die Softwaretechnik

IN0011: Einführung in die Theoretische Informatik ED

ED0136: Didaktik der Informatik 1

ED0134: Didaktik der Informatik 2

### Inhalt:

Stand der aktuellen Forschung zur Didaktik der Informatik mit Bezug zur Unterrichtspraxis; aktuelle Themen der Informatikdidaktik; Wissenschaftliche Methoden der Informatik; Grundzüge des Wissenschaftlichen Arbeitens; Fachdidaktik in der Unterrichtspraxis: Planung, Durchführung und Auswertung von Informatikunterricht, insbesondere fachgerechte Gestaltung von Lernarrangements.

### Lernergebnisse:

Die Teilnehmer können selbständig (evt. auch in Gruppenarbeit) eine Seminarausarbeitung zu einem anspruchsvolleren wissenschaftlichen Thema aus dem Gebiet der Didaktik der Informatik erarbeiten und diese präsentieren. Diese Aufgabe kann auch eigene Forschungsaktivitäten in geringem Umfang beinhalten. Sie

beherrschen die hierzu erforderlichen wissenschaftlichen Methoden und Präsentationstechniken; die Teilnehmer können die in den fachdidaktischen Lehrveranstaltungen erlernten Kenntnisse und Fertigkeiten im Informatikunterricht ihrer Schulart anwenden; insbesondere können Sie ihren Unterricht fachgerecht planen und durchführen. Sie sind in der Lage, erlebtes und/oder selbst gestaltetes Unterrichtsgeschehen kritisch zu reflektieren; Sie beherrschen geeignete Techniken, um die Lernfortschritte der Schülerinnen und Schüler festzustellen und zu bewerten.

**Lehr- und Lernmethoden:**

Im Seminar werden theoretische Inhalte zum wissenschaftlichen Arbeiten mittels Instruktion und in Kleingruppen erarbeitet. Die Arbeit der Studierenden wird durch Literaturrecherche, Datenerhebung, Datenauswertung praktisch ergänzt. Außerdem ist die Anfertigung einer Ausarbeitung und eine Präsentation im Rahmen des Vortrags notwendig. Im Rahmen des Schulpraktikums hospitieren die Studierenden Unterrichtseinheiten und halten selber Unterricht. Die gemachten Erfahrungen reflektieren Sie in einem Bericht.

**Medienform:**

Elektronische Präsentation;  
E-Learning System Moodle;  
Datenaustausch über Mahara;  
Betreuung mit Skype

**Literatur:**

Hubwieser P.: Didaktik der Informatik. 3. Auflage, Springer, Berlin, 2007; Weitere Quellen je nach Thema und aktueller Lage

**Modulverantwortliche(r):**

keine Angabe: [keine Angabe](#)

**Lehrveranstaltungen (Lehrform, SWS) Dozent(in):**

0000001230 Hauptseminar - Didaktik der Informatik (2SWS SE, WS 2020/21) [BF]  
Geldreich K

---

Weitere Informationen zum Modul und seiner Zuordnung zum Curriculum:

<https://campus.tum.de/tumonline/wbModHb.wbShowMHBRReadOnly?pKnotenNr=1301976>

Generiert am: 22.01.2021 23:06