UNIVERSITÄT BAYREUTH

Modulhandbuch

für Informatik Bachelor (LA Gym)
in der Fassung vom 15. Februar 2018 ÄS 20. Mai 2026

Fakultät für Mathematik, Physik und Informatik
Universität Bayreuth

Allgemeine Informationen und Lesehinweise

Ein zentraler Baustein des Bologna-Prozesses ist die Modularisierung der Studiengänge, das heißt die Umstellung des vormaligen Lehrveranstaltungssystems auf ein Modulsystem, in dem die Lehrveranstaltungen zu thematisch zusammenhängenden Veranstaltungsblöcken – also Modulen – gebündelt sind.

Dieses Modulhandbuch enthält die Beschreibungen aller Module, die im Studiengang angeboten werden. Das Modulhandbuch dient der Transparenz und versorgt Studierende, Studieninteressierte und andere interne und externe Adressatinnen und Adressaten mit Informationen über die Inhalte der einzelnen Module, ihre Qualifikationsziele sowie qualitative und quantitative Anforderungen.

Inhaltsverzeichnis und Index

Das Inhaltsverzeichnis gibt einen Überblick über die Modulbereiche und Module des Studiengangs. Die Klammerangabe nach dem Titel eines Moduls enthält den Zeitpunkt, zu dem dessen Beschreibung zuletzt aktualisiert wurde. Beispiel zur Notation: 24W steht für das Wintersemester 2024/25, 25S für das darauffolgende Sommersemester 2025.

Der Index am Ende des Modulhandbuchs listet alle Module des Studiengangs in alphabetischer Reihenfolge auf.

Modulbeschreibung

Die Beschreibung eines Moduls enthält dessen Lernziele, Lerninhalte und Leistungsnachweise. Sieht ein Modul mehrere Leistungsnachweise vor, ist deren Anteil an der Modulgesamtnote als Gewichtung angegeben. Eine detaillierte Beschreibung der Prüfungsformen, durch die die Leistungsnachweise erbracht werden, findet sich in § 9 der Allgemeinen Prüfungs- und Studienordnung für die Bachelor- und Masterstudiengänge an der Universität Bayreuth.

Durch Scannen oder Anklicken des QR-Code in der Beschreibung gelangt man zur Webseite des Moduls. Hierüber lassen sich die zum Modul gehörigen Lehrveranstaltungen einsehen und es wird angegeben, in welchen weiteren Studiengängen das Modul vorkommt.

Rechtsverbindlichkeit

Modulbeschreibungen dienen der Erhöhung der Transparenz und der besseren Orientierung über die Module eines Studiengangs. Rechtsverbindlich ist ausschließlich die einschlägige *(Fach-)Prüfungs- und Studienordnung.*

Inhaltsverzeichnis

Alphabetisches Verzeichnis befindet sich auf Seite 13

Fachwissenschaft (FW)
Fak127933: Konzepte der Programmierung (25W) 4
Fak127934: Rechnerarchitektur und Rechnernetze (25W) 5
Fak127935: Algorithmen und Datenstrukturen I (25W) 6
Fak127936: Betriebssysteme (25W) 7
Fak127951: Datenbanken und Informationssysteme I (25W) 8
Fak112537: LAI 911: Programmierpraktikum (16S) 9
Fak127986: Formale Grundlagen der Informatik (25W) 10
Fak112539: Softwarepraktikum für Lehramtsstudierende (20W) 11
Fachdidaktik (FD)
Fak112536: LAI 221: Informatik – Lehren und Lernen (16S) 12

Fak127933: Konzepte der Programmierung

Gültig ab: 01.10.2025

Lehrsprache	Dauer	Präsenzstudium (in Stunden)	Selbststudium (in Stunden)
Deutsch	Einsemestrig	90	150

Leistungspunkte (LP)	Turnus	Modulverantwortliche(r)
8	Wintersemester	Mayer, Ruben; Prof. Dr.

Leistungsnachweise

Titel:	Gewichtung:	LP:
Klausur oder mündliche Prüfung	1	8

Studien-/Prüfungsleistungen: Aktive Teilnahme an den Veranstaltungen und Selbststudium

Lernziele

Ziel der Veranstaltung ist, den Studierenden ein fundiertes Verständnis der Programmierung zu vermitteln, das im weiteren Studium als Fundament für die Informatik-Ausbildung dient. Dabei dient Java als Beispielsprache.
Der Vorkurs richtet sich insbesondere an Studierende ohne Programmiervorkenntnisse, wird aber allen Studierenden empfohlen. Er dient dazu, der fachlichen Diversität der Studierenden zu begegnen. Als Einstiegssprache wird Python verwendet.
Der Schwerpunkt liegt auf dem Erwerb von methodischen Kompetenzen: Durch das Verständnis fundamentaler Konzepte wie Kontroll- und Datenstrukturen, Methoden, Objektorientierung, Syntax, Typkonzept etc. sollen die Studierenden in die Lage versetzt werden, diese Konzepte bei der Umsetzung von Algorithmen in Programme einzusetzen und sich ferner in andere Programmiersprachen einzuarbeiten. Erste algorithmische Kompetenzen werden ebenfalls erworben. Sie legen die Grundlage für weiterführende Veranstaltungen (z.B. Algorithmen und Datenstrukturen).
Die Intensivübungen richten sich an Studierende, die die Modulprüfung im Sommersemester absolvieren und zur individuellen Förderung und Differenzierung.

Lerninhalte

Vorkurs: Grundbegriffe, Algorithmen, einfache Programme in Python
Einführung: Grundbegriffe
Algorithmen: wesentliche Eigenschaften, erste Beispiele
Programme: Umsetzung von Algorithmen in eine Programmiersprache
Syntax: EBNF, Ableitungsbäume, Syntaxdiagramme
Elementare Datentypen: ganze Zahlen, Gleitpunktzahlen, Zeichen, Wahrheitswerte
Ausdrücke: Syntax, Prioritäten, Auswertungsbäume
Anweisungen: Zuweisungen, Kontrollstrukturen, Flussdiagramme, strukturierte Programmierung
Methoden: Syntax, Aufruf, Ausführung, Aufrufstapel
Rekursion: Klassifikation von Rekursionsarten, Ausführung, Elimination von Rekursion
Strukturierte Datentypen: Arrays und Verbunde; Eigenschaften von Objekttypen, Referenzen
Objekte und Klassen: Grundbegriffe, Felder, Methoden, Konstruktoren, Klassen- vs. Objekteigenschaften, abstrakte Klassen
Vererbung: Einfachvererbung, Substituierbarkeit, Polymorphie, Vererbungsregeln
Schnittstellen: Konzept, Abgrenzung gegen abstrakte Klassen, Implementierung von Schnittstellen, Mehrfachvererbung auf Schnittstellen
Generizität: generische Datentypen, Abgrenzung von Generizität und Vererbung
Ausnahmebehandlung: Ausnahmeobjekte, Ausnahmebehandler, geschützte Blöcke
Funktionale Programmierung in Java
Inhalt der Intensivübung: Programmierung ausgewählter Kapitel der Vorlesung

Art und Umfang der Lehrveranstaltungen

Vorlesung (4 SWS)
Übungen (2 SWS)
Intensivübung (2 SWS, optional)
Vorkurs (1 SWS, optional)

Literatur:

K. Echtle, M. Goedicke: Lehrbuch der Programmierung, dpunkt.verlag, Heidelberg, 2000
H.P. Mössenböck: Sprechen Sie Java?, 5. Auflage, dpunkt.Verlag, Heidelberg, 2014
D.J. Barnes, M. Kölling: Objektorientierte Programmierung mit Java - Eine praxisnahe Einführung mit BlueJ, Pearson Studium, München (2003)
H. Balzert: Objektorientierte Programmierung mit Java 5, Spektrum Akademischer Verlag, Heidelberg (2005)
R. Sedgewick, K. Wayne: Einführung in die Programmierung mit Java, Pearson, München, 2011

zum Inhaltsverzeichnis

Fak127934: Rechnerarchitektur und Rechnernetze

Gültig ab: 01.10.2025

Lehrsprache	Dauer	Präsenzstudium (in Stunden)	Selbststudium (in Stunden)
Deutsch	Einsemestrig	90	150

Leistungspunkte (LP)	Turnus	Modulverantwortliche(r)
8	Wintersemester	Rauber, Thomas; Prof. Dr.

Leistungsnachweise

Titel:	Gewichtung:	LP:
Klausur oder mündliche Prüfung	9	7,2
semesterbegleitende Aufgaben	1	0,8

Lernziele

Das Ziel der Veranstaltung besteht in der Vermittlung grundlegender technologischer Kompetenz mit dem Schwerpunkt der Vermittlung von Kenntnissen des Aufbaus von Rechnersystemen mit Speicherhierarchie und Prozessoren. Vermittelt werden auch formale und algorithmische Kompetenzen, die zur Analyse und dem Entwurf digitaler Schaltkreise befähigen, sowie Design- und Realisierungskompetenzen zum Entwurf komplexer Schaltkreise. Durch Erlernen qualitativer Analyseverfahren zur Leistungsbewertung von Rechnersystemen und Rechnernetzen werden grundlegende methodische Kompetenzen im Bereich Rechnersysteme und Rechnernetze erworben, die Grundlagen für weiterführende Veranstaltungen legen.

Lerninhalte

Leistungsbewertung von Rechnern und grundsätzlicher Rechneraufbau
Maschinensprachen als Schnittstelle zwischen Hardware und Software
Zahlendarstellungen und Rechnerarithmetik
Entwurf digitaler Schaltkreise
Kombinatorische Schaltungen
Konstruktion von Speicherelementen
Speicherorganisation und Prozessorganisation,
Grundlagen und Leistungsbewertungen von Rechnernetzen
Schichtenprotokolle und Kommunikationsablauf
Wichtige Protokolle von Verbindungsschicht, Netzwerkschicht und Transportschicht

Art und Umfang der Lehrveranstaltungen

Vorlesung (4 SWS)
Übungen (2 SWS)

Literatur:

Patterson/Hennessy: Computer Organization & Design, Morgan Kaufmann, 6. Auflage, 2020
Hennessy/Patterson: Computer Architecture, Morgan Kaufmann, 7. Auflage, 2026
Oberschelp/Vossen: Rechneraufbau und Rechnerstrukturen, Oldenbourg Verlag, 10. Auflage, 2006
Kurose/Ross: Computer Networking, Addison Wesley, 8. Auflage, 2021
Peterson, Davie: Computer Networks, Morgan Kaufmann, 6. Auflage, 2021.

zum Inhaltsverzeichnis

Fak127935: Algorithmen und Datenstrukturen I

Gültig ab: 01.10.2025

Lehrsprache	Dauer	Präsenzstudium (in Stunden)	Selbststudium (in Stunden)
Deutsch	Einsemestrig	90	150

Leistungspunkte (LP)	Turnus	Modulverantwortliche(r)
8	Sommersemester	Knauer, Christian; Prof. Dr.

Leistungsnachweise

Titel:	Gewichtung:	LP:
Klausur	1	8

Studien-/Prüfungsleistungen: Aktive Teilnahme an den Veranstaltungen und Selbststudium

Voraussetzungen

Inhalte der Module
- Konzepte der Programmierung
- Formale Grundlagen der Informatik

Lernziele

Die Studenten sollen lernen, Daten zu strukturieren und dynamisch zu repräsentieren. Wichtig ist hierbei die enge Verknüpfung dieser Datenstrukturen und der hierauf angewandten Algorithmen. Ein weiteres Ziel ist die Vermittlung von Kenntnissen zur Komplexitätsanalyse von Algorithmen (methodische Kompetenz).
In der Fragestunde können Lehrinhalte beim Dozenten spezifisch nachgefragt und individuell nachgearbeitet werden. Sie dient der individuellen Förderung.

Lerninhalte

Listen, Keller, Such- und Sortierverfahren, binäre Bäume, Suchbäume (AVL), Graphen, Hash-Verfahren, Komplexität von Algorithmen, Algorithmentheorie.

Art und Umfang der Lehrveranstaltungen

Vorlesung (4 SWS)
Übung (2 SWS)
Fragestunde (1 SWS, optional)

Literatur:

Goodrich, Tamassia: “Data Structures & Algorithms in Java” (4th Ed.), 2006.
Saake, Sattler: “Algorithmen und Datenstrukturen – Eine Einführung mit Java” (3. Aufl.), 2006.
Cormen, Leiserson, Rivest, Stein: “Introduction to Algorithms” (2nd Ed.), 2001.

zum Inhaltsverzeichnis

Fak127936: Betriebssysteme

Gültig ab: 01.10.2025

Lehrsprache	Dauer	Präsenzstudium (in Stunden)	Selbststudium (in Stunden)
Deutsch	Einsemestrig	45	105

Leistungspunkte (LP)	Turnus	Modulverantwortliche(r)
5	Wintersemester	Henrich, Dominik; Prof. Dr.

Leistungsnachweise

Titel:	Gewichtung:	LP:
Klausur oder mündliche Prüfung zu Betriebssysteme	17	4,25
semesterbegleitende Aufgaben zu Betriebssyteme	3	0,75

Studienleistung: Aktive Teilnahme an den Veranstaltungen und Selbststudium
Prüfungsleistung: Siehe FPSO

Voraussetzungen

- Konzepte der Programmierung
- Rechnerarchitektur und Rechnernetze
- Algorithmen und Datenstrukturen I

Lernziele

Lernziele des Moduls sind das Verständnis des grundsätzlichen Aufbaus von Betriebssystemen, das Verständnis der eingesetzten Verfahren, sowie das Lernen der sinnvollen Auswahl und des Einsatz von Betriebssystemen. Es werden keine Einschränkung auf ein bestimmtes Betriebssystem vorgenommen und auch keine Implementierungsdetails vermittelt. Allgemein werden Methoden zur effizienten Verwaltung von zeitlichen bzw. räumlichen Ressourcen vermittelt.

Lerninhalte

Einleitung: Definition, Schnittstellen, Historie, Aufbau;
Prozessverwaltung: Prozesse/Threads, Prozesssynchronisation, -scheduling, -kommunikation;
Speicherverwaltung: Speicherbelegung, -adressierung, -seiten, Segmentierung, Caches, Schutz;
Dateiverwaltung: Dateisysteme, -namen, -attribute, -funktionen, -organisation;
Ein-/Ausgabeverwaltung: E/A-Aufgaben, Gerätemodelle, Treiber;
Systemsicherheit: Kryptographie, Authentifikation, Angriffe, Schutz;

Art und Umfang der Lehrveranstaltungen

Vorlesung (2 SWS)
Übung (1 SWS)

Anmerkungen

Medienformen: Multimedia-Präsentation als Vortrag und Ausdruck, Übungsblätter mit Korrektur, Tafelübungen

Literatur:

Tanenbaum A. S.: „Moderne Betriebssysteme“. 2. überarbeitete Auflage, München, Pearson Studium, 2002. –> 82/ST 260 T164 M6(3)+3. Dritte Auflage von 2009

zum Inhaltsverzeichnis

Fak127951: Datenbanken und Informationssysteme I

Gültig ab: 01.10.2025

Lehrsprache	Dauer	Präsenzstudium (in Stunden)	Selbststudium (in Stunden)
Deutsch	Einsemestrig	90	150

Leistungspunkte (LP)	Turnus	Modulverantwortliche(r)
8	Sommersemester	Jablonski, Stefan; Prof. Dr.-Ing.

Leistungsnachweise

Titel:	Gewichtung:	LP:
Klausur oder mündliche Prüfung	1	8

Studien-/Prüfungsleistungen: Aktive Teilnahme an den Veranstaltungen und Selbststudium

Lernziele

Ziel ist die Vermittlung grundlegender Kenntnisse zum Entwurf von (relationalen) Datenbanken. Die Studierenden sollen Analyse-, Entwurfs und Realisierungskompetenzen vermittelt bekommen, so dass sie selbständig eine Anwendungssituation analysieren und darauf aufbauend ein datenbankgestützte Anwendungen entwickeln können.
Daneben sollen Grundkenntnisse bezüglich des Aufbaus und des Betriebs von Datenbanksystemen vermittelt werden, so dass die Studierenden einen prinzipiellen Einblick in die Technologie von Datenbanksystemen bekommen.
Über den Übungsbetrieb sollen die Studierenden den praktischen Umgang mit Datenbanken und deren Anwendungen erlernen.
In den Intensivübungen werden darüber hinaus programmiertechnische Fähigkeiten vermittelt und Studierende individuell gefördert.

Lerninhalte

Entwurf von Datenbanksystemen: Aufbau konzeptioneller Schemata (Von Entity-Relationship-Diagrammen zu Relationen), Normalisierung, Relationenalgebra, Einführung in SQL, Verwendung von Datenbanksystemen (SQL als DB-Schnittstelle), Objektrelationale Datenbanksysteme; Aufbau von Datenbanksystemen (Architektur), Einführung ins Transaktionsmanagement; Aufbau von Informationssystemen (Arten von Informationssystemen), Anwendungen von Datenbanken in den Bereichen Bio-, Ingenieur- und Umweltinformatik; Vorstellung von Beispielen und Fallstudien.
Inhalt der Intensivübung: Programmierung ausgewählter Kapitel der Vorlesung.

Art und Umfang der Lehrveranstaltungen

Vorlesung (4 SWS)
Übung (2 SWS)
Intensivübung (2 SWS, optional)

Literatur:

Härder, T.; Rahm, E.: Architektur von Datenbanksystemen. Springer-Verlag
Gray, J.; Reuter, A.: Transaction Systems. Morgan Kaufman
Conolly, T.; Begg, C.: Database Systems: A Practical Approach to Design, Implementation and Management. Addison-Wesley Longman

zum Inhaltsverzeichnis

Fak112537: LAI 911: Programmierpraktikum

Gültig ab: 01.04.2016

Lehrsprache	Dauer	Präsenzstudium (in Stunden)	Selbststudium (in Stunden)
Deutsch	Einsemestrig		

Leistungspunkte (LP)	Turnus	Modulverantwortliche(r)
5	Sommersemester	Korch, Matthias; apl. Prof. Dr.

Leistungsnachweise

Titel:	Gewichtung:	LP:
Programmierpraktikum	1	5

150 h gesamt (60 h Präsenz, 90 h Vorbereitung und Übungsbearbeitung)

Voraussetzungen

Keine

Lernziele

Vermittlung grundlegender Programmiertechniken; Einführung in die funktionale Programmierung; Einführung in die Programmierung mit Skriptsprachen

Lerninhalte

Vermittlung grundlegender Programmiertechniken; Einführung in die funktionale Programmierung; Einführung in die Programmierung mit Skriptsprachen

Art und Umfang der Lehrveranstaltungen

Praktikum

zum Inhaltsverzeichnis

Fak127986: Formale Grundlagen der Informatik

Gültig ab: 01.10.2025

Lehrsprache	Dauer	Präsenzstudium (in Stunden)	Selbststudium (in Stunden)
Deutsch	Einsemestrig	90	150

Leistungspunkte (LP)	Turnus	Modulverantwortliche(r)
8	Sommersemester	Stehn, Fabian; Dr.

Leistungsnachweise

Titel:	Gewichtung:	LP:
Klausur	1	8

Studien-/Prüfungsleistungen: Aktive Teilnahme an den Veranstaltungen und Selbststudium

Voraussetzungen

Umgang mit Gleichungssystemen, insb. Gauß-Verfahren

Lernziele

Die Studierenden kennen die grundlegenden Definitionen und Eigenschaften der Logiken, die in der Vorlesung behandelt wurden. Sie können die Verfahren aus der Vorlesung auf Beispiele anwenden. Sie sind in der Lage, umgangssprachliche Texte formal zu modellieren.
Die Studierenden kennen formale Methoden aus dem Bereich der diskreten Mathematik. Sie kennen Anwendungen dieser Methoden auf Probleme der Informatik. Die Studierenden können die vorgestellten Methoden selber einsetzen. Sie sind in der Lage zu erkennen, wann ähnliche Situationen für die Anwendung der bekannten Verfahren vorliegen. Die Studierenden können einfache formale Beweise aus dem Bereich der diskrete Mathematik durchführen.
In der Fragestunde können Lehrinhalte beim Dozenten spezifisch nachgefragt und individuell nachgearbeitet werden.
Im Mathematik-Vorkurs werden die wichtigsten mathematischen Grundlagen aus der gymnasialen Oberstufe wiederholt und vertieft. Ziel ist, potentielle Unterschiede in den Vorkenntnissen der Studenten auszugleichen und eventuell vorhandene Wissenslücken aufzufüllen.

Lerninhalte

- Diskrete Strukturen:
Beweismethoden (Direkter Beweis, Indirekte Beweise: Kontraposition und Widerspruch, vollständige Induktion, Ringschluss, Beweis durch Zeugen), Mengen, Relationen, Funktionen, Landau-Symbole, Abzählbarkeit und Überabzählbarkeit, elementare Zahlentheorie, ggT, kgV, Fundamentalsatz der Zahlentheorie, Kombinatorik.
- Logik und Modellierung:
Aussagenlogik, Modallogik und Prädikatenlogik. Syntax, Semantik und Eigenschaften. Übungen in Modellierung.
- Mathematik-Vorkurs:
Aussagen- und Prädikatenlogik; Lösen von Gleichungen; Mengenlehre; Funktionen; Komplexe Zahlen; Kombinatorik; Relationen

Art und Umfang der Lehrveranstaltungen

Vorlesung 'Diskrete Strukturen' (2 SWS)
Übung 'Diskrete Strukturen' (1 SWS)
Vorlesung 'Logik und Modellierung' (2 SWS)
Übung 'Logik und Modellierung' (1 SWS)
Fragestunde (1 SWS, optional)
Mathematik-Vorkurs (1 SWS als Block vor Vorlesungsbeginn, optional)

Literatur:

Diskrete Strukturen:
- Kurt Ullrich Witt: Mathematische Grundlagen der Informatik
- Kurt Ullrich Witt: Elementare Kombinatorik für die Informatik
Logik und Modellierung:
- Martin Kreuzer, Stefan Kühling: Logik für Informatiker
- Uwe Schöning: Logik für Informatiker

zum Inhaltsverzeichnis

Fak112539: Softwarepraktikum für Lehramtsstudierende

Gültig ab: 01.10.2020

Lehrsprache	Dauer	Präsenzstudium (in Stunden)	Selbststudium (in Stunden)
Deutsch	Einsemestrig	60	120

Leistungspunkte (LP)	Turnus	Modulverantwortliche(r)
7	jedes Semester	Ehmann, Matthias; Dr.

Leistungsnachweise

Titel:	Gewichtung:	LP:
Softwarepraktikum für Lehramtsstudierende	1	7

Modulprüfung

Voraussetzungen

INF 107 - Konzepte der Programmierung
INF 109 - Algorithmen und Datenstrukturen I

Lernziele

Die Studierenden entwickeln unter Anleitung kleinere Softwaresysteme in kleinen Gruppen. Im Vordergrund steht der Erwerb von algorithmischen, Design- und Realisierungskompetenzen. Die Studierenden sollen in die Lage versetzt werden, Software beschränkten Umfangs und beschränkten Schwierigkeitsgrads syste-matisch zu entwickeln (methodische Kompetenz) sowie die von ihnen erarbeitete Lösung zu präsentieren (kommunikative Kompetenz).

Lerninhalte

Probleme werden analysiert, Anforderungen definiert, ein Systementwurf erstellt, und die Komponenten des Systementwurfs werden implementiert und getestet. Hinzu kommt die Präsentation der Lösungskonzepte.

Anmerkungen

Präsentation der Aufgabenstellung und der Lösungskonzepte

Literatur:

Wolfram E. Rossig: Wissenschaftliches Arbeiten, Rossig Verlag, 6. Auflage, 2006

zum Inhaltsverzeichnis

Fak112536: LAI 221: Informatik – Lehren und Lernen

Gültig ab: 01.04.2016

Lehrsprache	Dauer	Präsenzstudium (in Stunden)	Selbststudium (in Stunden)
Leistungspunkte (LP) 4	**Turnus**	**Modulverantwortliche(r)** Ehmann, Matthias; Dr.	

Leistungsnachweise

Titel:	Gewichtung:	LP:
Modulprüfung Informatik – Lehren und Lernen	4	4

Lernziele

Erläutern und Anwenden der Konzepte des Lehrens und Lernens im Fach Informatik Beherrschen der Denkweisen und Methoden der Informatik und ihre Übertragung auf den Schulunterricht Sicherer Umgang mit Medien im Informatikunterricht

zum Inhaltsverzeichnis

Index

Fak127935: Algorithmen und Datenstrukturen I (25W) 6
Fak127936: Betriebssysteme (25W) 7
Fak127951: Datenbanken und Informationssysteme I (25W) 8
Fak127986: Formale Grundlagen der Informatik (25W) 10
Fak127933: Konzepte der Programmierung (25W) 4
Fak112536: LAI 221: Informatik – Lehren und Lernen (16S) 12
Fak112537: LAI 911: Programmierpraktikum (16S) 9
Fak127934: Rechnerarchitektur und Rechnernetze (25W) 5
Fak112539: Softwarepraktikum für Lehramtsstudierende (20W) 11