



LUDWIG-
MAXIMILIANS-
UNIVERSITÄT
MÜNCHEN

UNIVERSITÄT MÜNCHEN
INSTITUT FÜR INFORMATIK



Modulhandbuch

Lehramt Gymnasium (INF-LGY)

72 ECTS-Punkte

Auf der Basis der Prüfungs- und Studienordnung
vom 25.1.2010 mit Änderungen vom 16.2.2012 und 23.4.2012

Version (24.01.2018)

Der Studiengang

Das Studium Lehramt Gymnasium, Teilgebiet Informatik, bereitet auf die berufliche Praxis auf dem Gebiet der Informatik in lehrbezogener Tätigkeit vor, insbesondere für das Gymnasium. Das Ziel der Ausbildung ist es, die Grundlagen des Faches in theoretischer und praktischer Hinsicht zu erarbeiten und auf dieser Basis in der Lage zu sein, die fachlichen Inhalte schülergerecht zu vermitteln.

Es soll die Befähigung entwickelt werden, vielfältige Probleme der Informationsverarbeitung selbstständig zu erkennen und zu lösen. Das Studium vermittelt einerseits Erkenntnisse und Methoden in den zentralen Gebieten der Informatik auf der Basis formaler Grundlagen, andererseits die Kompetenz, diese Erkenntnisse didaktisch fundiert weiterzugeben. Nach Abschluss der Ausbildung sollen Kenntnisse über Eigenschaften und formale Beschreibungsmöglichkeiten von Informationsverarbeitungsprozessen sowie über Strukturen und Wirkungsweisen von Informationsverarbeitungssystemen vorhanden sein. Besondere Bedeutung kommt der Fähigkeit zu, sich auf wechselnde Aufgabengebiete einstellen zu können, sich den sich wandelnden Bedingungen der Praxis der Informationsverarbeitung anpassen zu können und diesen Wandel aktiv mitzugestalten. Der fachwissenschaftliche Hintergrund ermöglicht es, Inhalte auszuwählen und diese hinsichtlich der Eignung für einen schülergerechten Unterricht zu beurteilen. Des Weiteren werden Planung, Organisation und Durchführung von Informatikunterricht vermittelt, sowie Kenntnis, Analyse und Bewertung von Lehr- und Lernprozessen.

Der Studiengang besteht aus zwei Teilen: dem fachdidaktischen und dem fachwissenschaftlichen Teil. Kombiniert werden kann dieser Studiengang an der LMU mit den Fächern Mathematik, Englisch oder Wirtschaftswissenschaften. Insgesamt dauert das Lehramtsstudium 9 Semester. Die Studierenden müssen in dieser Zeit 93 ECTS-Punkte im fachwissenschaftlichen Bereich erwerben, davon müssen mindestens 15 Leistungspunkte aus dem Gebiet Theoretische Informatik und Algorithmen und Datenstrukturen sein, mindestens 20 Punkte aus dem Gebiet Datenbanksysteme und Softwaretechnologie, mindestens 10 aus dem Gebiet der Technischen Informatik und mindestens 15 aus dem Gebiet Praktische Softwareentwicklung, einschließlich je eines Praktikums zur Praktischen Programmierung und zur planmäßigen Entwicklung eines Softwaresystems. Zusätzlich werden insgesamt 12 ECTS aus dem fachdidaktischen Bereich erworben, darunter ein Praktikum zu Informatiksystemen aus fachdidaktischer Sicht. Desweiteren benötigen die Studierenden ECTS-Punkte aus einer Hausarbeit, einem Seminar zum studienbegleitenden Praktikum und einem pädagogisch-didaktischen Schulpraktikum, die aber nicht notwendigerweise im Fach Informatik erworben werden müssen. Laut Studienordnung müssen außerdem weitere 6 ECTS-Punkte aus einem fachdidaktischen oder fachwissenschaftlichen Bereich erworben werden.

Studienbeginn: WiSe, SoSe.

Inhaltsverzeichnis

1 Erklärungen	4
2 Reguläre Module	5
2.1 P 1: Einführung in die Programmierung (INF-EiP)	6
2.2 P 2: Fachdidaktik I (INF-FD1)	9
2.3 P 3: Programmierung und Modellierung (INF-ProMo)	11
2.4 P 4: Rechnerarchitektur (INF-RA)	14
2.5 P 5: Softwareentwicklungspraktikum (INF-SEP)	17
2.6 P 6: Formale Sprachen und Komplexität (INF-FSK)	19
2.7 P 7: Algorithmen und Datenstrukturen (INF-AIDs)	22
2.8 P 8: Datenbanksysteme I (INF-DBSI)	27
2.9 P 9: Softwaretechnik (INF-SWT)	29
2.10 P 10: Vertiefende Themen aus der theoretischen Informatik (INF-VT)	31
2.11 P 11: Vertiefende Themen aus der technischen Informatik (INF-VT)	32
2.12 P 12: Informatikseminar (INF-Sem)	33
2.13 P 13.1: Grundlagen der Fachdidaktik II (INF-FD2-GY)	35
2.14 P 13.2 und 13.3: Fachdidaktisches Praktikum und Seminar (INF-FDPS-GY)	37
2.15 P 14: Vertiefende Themen aus Datenbanksysteme und Softwaretechnologie (INF-VT)	39
2.16 P 15: Praktische Programmierung (INF-PP)	40
3 Vertiefende Themen	42
3.1 VT 1: Mobilkommunikation (INF-MK)	43
3.2 VT 2: Sensornetze (INF-SN)	45
3.3 VT 3: Rechnerarchitektur 2 (INF-RA2)	47
3.4 VT 4: Praktikum Innovative Mobile Business Applications (INF-MBA)	49
3.5 VT 5: Praktikum für iOS Entwicklung (INF-IOS)	52
3.6 VT 6: Automotive Praktikum (INF-AP)	55
3.7 VT 7: Geistiges Eigentum für die Informationstechnologie (INF-IPIT)	57
3.8 VT 8: Mobile und Verteilte Systeme (INF-MVS)	60
3.9 VT 9: Methoden des Software Engineering (INF-MSE)	63
3.10 VT 10: Formale Techniken in der Software-Entwicklung (INF-FTS)	65
3.11 VT 11: Software Engineering für spezielle Anwendungsgebiete (INF-SEspA)	67

1 Erklärungen

CP	Credit Points, ECTS-Punkte
ECTS	European Credit Transfer and Accumulation System
h	Stunden
SS	Sommersemester
WS	Wintersemester
SWS	Semesterwochenstunden
GOP	Grundlagen- und Orientierungsprüfung

1. Bitte beachten Sie: Das Modulhandbuch dient einer Orientierung für Ihren Studienverlauf. Für verbindliche Regelungen konsultieren Sie bitte ausschließlich die Prüfungs- und Studienordnung in ihrer jeweils geltenden Fassung. Diese finden Sie auf www.lmu.de/studienangebot unter ihrem jeweiligen Studiengang.

2. Die Beschreibung der zugeordneten Modulteile erfolgt hinsichtlich der jeweiligen Angaben zu ECTS-Punkten folgendem Schema: Nicht eingeklammerte ECTS-Punkte werden mit Bestehen der zugehörigen Modulprüfung oder Modulteilprüfung vergeben. Eingeklammerte ECTS-Punkte dienen lediglich der rechnerischen Zuordnung.

3. Module, deren Kennzeichnung mit P anfängt sind Pflichtmodule.

Module, deren Kennzeichnung mit WP anfängt sind Wahlpflichtmodule.

Module, deren Kennzeichnung mit A anfängt sind zusätzliche Angebote, die nicht regelmäßig stattfinden, aber als *Vertiefende Themen* angerechnet werden können.

2 Reguläre Module

Die folgenden Module entsprechen der Prüfungs- und Studienordnung. Falls in der Aufzählung der Pflichtmodule oder Wahlpflichtmodule einzelne Nummern fehlen, handelt es sich um Platzhalter für Module zu vertiefenden Themen.

2.1 P 1: Einführung in die Programmierung (INF-EiP)

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tu- rus	Präsenzzeit	Selbst- studium	ECTS
Vorlesung	Vorlesung zur Einführung in die Programmierung	WiSe	60 h (4 SWS)	120 h	6 CP
Übung	Übungen zur Einführung in die Programmierung	WiSe	30 h (2 SWS)	60 h	3 CP

Im Modul müssen insgesamt 9 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 6 Wochenstunden. Inklusiv Selbststudium sind etwa 270 Stunden aufzuwenden.

Art des Moduls Pflichtmodul mit Pflichtveranstaltungen

Verwendbarkeit Dieses Modul wird in folgenden Studiengängen angeboten

- INF-B-120: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 60 ECTS-Punkten
- INF-B-150: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 30 ECTS-Punkten
- INF-B-180-CL: Bachelorstudiengang Informatik plus Computerlinguistik
- INF-B-180-MA: Bachelorstudiengang Informatik plus Mathematik
- INF-B-180-STAT: Bachelorstudiengang Informatik plus Statistik
- INF-LGY: Lehramt Gymnasium
- INF-LRS: Lehramt Realschule
- INF-NF-15: Nebenfach: Informatik für Bachelorstudiengänge
- MINF-B-180: Bachelorstudiengang Medieninformatik

Teilnahmevoraussetzungen keine

Zeitpunkt im Studienverlauf 1. Semester

Dauer Das Modul erstreckt sich über 1 Semester.

Art der Bewertung benotet

Form der Modulprüfung Klausur (90-180 Minute) oder mündlich (15-30 Minute)
 Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine
 GOP (Grundlagen und Orientierungsprüfung), auch für INF-B-120, INF-B-150, INF-B-180-CL, INF-B-180-MA, INF-B-180-STAT, INF-NF-15, MINF-B-180

Modul- verantwort- licher	Prof. Dr. Hans Jürgen Ohlbach
--	-------------------------------

Anbieter	Ludwig-Maximilians-Universität München Fakultät für Mathematik, Informatik und Statistik Institut für Informatik Kerninformatik
-----------------	--

Unterrichts- sprache(n)	Deutsch
------------------------------------	---------

Inhalte

Dieses Modul gibt eine Einführung in die imperative, objekt-orientierte und nebenläufige Programmierung anhand einer höheren Programmiersprache, z.B. Java. Neben Kenntnissen in der Programmierung werden allgemeine Grundlagen, Konzepte, Methoden und Techniken zur Darstellung, Strukturierung und Verarbeitung von Daten sowie zur Entwicklung von Algorithmen behandelt. Dabei wird auf begriffliche Klarheit und präzise mathematische Fundierung mit formalen Methoden Wert gelegt.

Im Einzelnen werden vermittelt:

- Grundbegriffe zu Programmen und ihrer Ausführung,
- Syntax von Programmiersprachen und ihre Beschreibung,
- Grunddatentypen und imperative Kontrollstrukturen,
- Komplexität und Korrektheit imperativer Programme,
- Rekursion,
- Einfache Sortierverfahren,
- Einführung in den objekt-orientierten Programmentwurf,
- Klassen, Schnittstellen und Pakete,
- Vererbung und Ausnahmebehandlung,
- Objektorientierte Realisierung von Listen- und Baumstrukturen,
- Grundkonzepte der nebenläufigen Programmierung: Threads, Synchronisation und Verklemmung,
- Einführung in UML-Diagramme.
- Benutzung einer Entwicklungsumgebung, derzeit Eclipse.

Literaturhinweise

Es gibt eine Vielzahl von einführenden Büchern zur Informatik und insbesondere zu Java.

Ein umfangreiches Buch über Java, welches es auch online gibt ist:

- Java ist auch eine Insel, von Christian Ullenboom, Gilileo Computing, ISBN = 978-3-8362-1802-3

Ein leichteres einführendes Buch ist

- Java kompakt, von Hözl, Read und Wirsing, Springer Vieweg, ISBN 978-3-642-28503-5

Das Modul besteht aus einer Vorlesung mit Übungen in kleinen Gruppen. Die in der Vorlesung besprochenen Inhalte werden im Übungsteil anhand von praktischen Anwendungen eingeübt.

Qualifikationsziele

Die Studierenden werden in die Lage versetzt, Lösungen für kleinere und überschaubare Probleme algorithmisch umzusetzen und mit einer höheren Programmiersprache als ausführbare Programme zu realisieren. Die Benutzung einer Entwicklungsumgebung wie Eclipse fördert die Professionalisierung. Des weiteren entwickeln die Studierenden ein Verständnis für die allgemeinen Prinzipien der Programmierung und der Programmiersprachen, das den Grundstein dafür legt, dass die Studierenden sich (nach weiteren Erfahrungen im Laufe des Studiums) in beliebige Programmiersprachen schnell und präzise einarbeiten können.

2.2 P 2: Fachdidaktik I (INF-FD1)

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tur- nus	Präsenzzeit	Selbst- studium	ECTS
Vorlesung	Grundlagen der Fachdidaktik I	SoSe	30 h (2 SWS)	90 h	3 CP

Im Modul müssen insgesamt 3 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 2 Wochenstunden. Inklusive Selbststudium sind etwa 90 Stunden aufzuwenden.

**Art des Mo-
duls** Pflichtmodul mit Pflichtveranstaltungen

**Verwendbar-
keit** Dieses Modul wird in folgenden Studiengängen angeboten
- INF-LGY: Lehramt Gymnasium
- INF-LRS: Lehramt Realschule

**Teilnahme-
voraus-
setzungen** keine

**Zeitpunkt
im Studien-
verlauf** 4. Semester (INF-LRS), 4./5. Semester (INF-LGY)

Dauer Das Modul erstreckt sich über 1 Semester.

**Art der Be-
wertung** benotet

**Form der
Modulprüfung** Klausur (60-120 Minute) oder mündlich (20-40 Minute) oder Hausarbeit (30-40 Seiten)
Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine

**Modul-
verantwort-
licher** Prof. PhD Martin Hofmann

Anbieter Ludwig-Maximilians-Universität München
Fakultät für Mathematik, Informatik und Statistik
Institut für Informatik

**Unterrichts-
sprache(n)** Deutsch

Inhalte

Die Studierenden beschäftigen sich mit allgemeinen didaktischen und fachdidaktischen Theorien, Konzeptionen und Forschungsfragen der Informatik. Eine Grundlage dafür ist die Kenntnis der Bildungsaufgaben, Lernziele und Lernbedingungen des Fachs Informatik in den einzelnen Schularten und dessen Beitrag für die Erfüllung der fächerübergreifenden Bildungs- und Erziehungsaufgaben. Eine Ergänzung bildet der Überblick über Geschichte und Stellung des Fachs Informatik im Fächerkanon. Spezielle Themen sind:

- Bildungs- und lerntheoretische Ansätze der Didaktik (Klafki, Berliner Modell),
- Curriculare Didaktik und Lernzielorientierung,
- Lernzieltaxonomien,
- Fundamentale Ideen der Informatik,
- Bildungsstandards,
- Repräsentationsebenen,
- Geschichte der Schulinformatik,
- Umgang mit Fehlern,
- Bewertung von Schülerarbeiten.

Das Modul besteht aus einer Vorlesung, in der Diskussionen und praktische Übungen einen festen Bestandteil haben; zusätzlich findet eine Übung (2 SWS) statt, die auf freiwilliger Basis besucht werden kann.

Qualifikationsziele

Die Studierenden kennen Aufgaben der Fachdidaktik und kennen didaktische Theorien und Fragestellungen, die sie auf den Informatikunterricht anwenden können. Sie können fachliche Fragestellungen analysieren und innerhalb und außerhalb der Schule kommunizieren und Bildungs- und Erziehungsaufgaben des Fachs Informatik diskutieren.

2.3 P 3: Programmierung und Modellierung (INF-ProMo)

Bemerkungen

Das vorherige Studium des Moduls *Einführung in die Programmierung* ist nützlich aber nicht zwingend. Anfang

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tu- rus	Präsenzzeit	Selbst- studium	ECTS
Vorlesung	Vorlesung zur Programmierung und Modellierung	SoSe	30 h (2 SWS)	30 h	2 CP
Übung	Übungen zur Programmierung und Modellierung	SoSe	45 h (3 SWS)	75 h	4 CP

Im Modul müssen insgesamt 6 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 5 Wochenstunden. Inklusiv Selbststudium sind etwa 180 Stunden aufzuwenden.

Art des Moduls	Pflichtmodul mit Pflichtveranstaltungen (INF-B-150, INF-B-180-CL, INF-B-180-MA, INF-B-180-STAT, INF-LGY, MINF-B-180), Wahlpflichtmodul mit Pflichtveranstaltungen (INF-NF-30, INF-NF-60)
-----------------------	--

Verwendbarkeit	Dieses Modul wird in folgenden Studiengängen angeboten - INF-B-150: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 30 ECTS-Punkten - INF-B-180-CL: Bachelorstudiengang Informatik plus Computerlinguistik - INF-B-180-MA: Bachelorstudiengang Informatik plus Mathematik - INF-B-180-STAT: Bachelorstudiengang Informatik plus Statistik - INF-LGY: Lehramt Gymnasium - INF-NF-30: Nebenfach: Informatik für Bachelorstudiengänge - INF-NF-60: Nebenfach: Informatik für Bachelorstudiengänge - MINF-B-180: Bachelorstudiengang Medieninformatik
-----------------------	--

Teilnahmevoraussetzungen	keine
---------------------------------	-------

Zeitpunkt im Studienverlauf	2. Semester
------------------------------------	-------------

Dauer	Das Modul erstreckt sich über 1 Semester.
--------------	---

Art der Bewertung	benotet
--------------------------	---------

Form der Modulprüfung Klausur (90-180 Minute) oder mündlich (15-30 Minute)
Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine
GOP (Grundlagen und Orientierungsprüfung), auch für INF-B-150, INF-B-180-CL, INF-B-180-MA, INF-B-180-STAT, INF-NF-30, INF-NF-60, MINF-B-180

Modulverantwortlicher Prof. PhD Martin Hofmann

Anbieter Ludwig-Maximilians-Universität München
Fakultät für Mathematik, Informatik und Statistik
Institut für Informatik
Kerninformatik

Unterrichtssprache(n) Deutsch

Inhalte

Dieses Modul führt in die grundlegenden Prinzipien der Programmierung und der Datenmodellierung mit einer funktionalen Programmiersprache ein (derzeit Haskell). Dabei wird auf begriffliche Klarheit und präzise mathematische Fundierung mit formalen Methoden Wert gelegt.

Die Themen sind unter anderem:

- Funktionsbegriff und Basistypen,
- Rekursion und Terminierung,
- Benutzerdefinierte Datentypen,
- Polymorphie, Typklassen, Module,
- Funktionen höherer Ordnung und Currying,
- Typen, Typprüfung, Typinferenz,
- Pattern Matching,
- Verzögerte Auswertung, Striktheit
- Ein- und Ausgaben und andere Seiteneffekte.

Literaturhinweise

- Miran Lipovača, "Learn You a Haskell for Great Good!", No Starch Press, 2011, ISBN 1-59327-283-9, kostenlose online-version verfügbar,
- Graham Hutton, "Programming in Haskell", Cambridge University Press, 2007, ISBN 0-52169269-5,
- Bryan O'Sullivan, Don Stewart, John Goerzen, "Real World Haskell", O'Reilly, November 2008, ISBN: 0-59651498-0, kostenlose online-version verfügbar,
- Simon Thompson, "Haskell: The Craft of Functional Programming", Second Edition, Addison-Wesley, 1999. ISBN 0-201-34275-8,
- Paul Hudak, John Peterson, Joseph Fasel, "A Gentle Introduction To Haskell", 2000, kostenloses online Tutorial.

Das Modul besteht aus einer Vorlesung mit Übungen in kleinen Gruppen. Die in der Vorlesung besprochenen Inhalte werden im Übungsteil anhand von praktischen Anwendungen eingeübt.

Qualifikationsziele

Das Modul zielt auf die Vermittlung des Folgenden:

- Beherrschung von grundlegenden Konzepten der (allgemeinen sowie deklarativen) Programmierung.
- Fähigkeit, kleine Algorithmen funktional zu programmieren und diese im Vergleich mit imperativen Lösungen zu bewerten.
- Vorbereitung auf die zukünftige Entwicklung von Programmiersprachen.

Bemerkungen

Das vorherige Studium des Moduls *Einführung in die Programmierung* ist nützlich aber nicht zwingend.

2.4 P 4: Rechnerarchitektur (INF-RA)

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tur- nus	Präsenzzeit	Selbst- studium	ECTS
Vorlesung	Vorlesung zur Rechnerarchitektur	SoSe	45 h (3 SWS)	45 h	3 CP
Übung	Übungen zur Rechnerarchitektur	SoSe	30 h (2 SWS)	60 h	3 CP

Im Modul müssen insgesamt 6 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 5 Wochenstunden. Inklusive Selbststudium sind etwa 180 Stunden aufzuwenden.

Art des Moduls	Pflichtmodul mit Pflichtveranstaltungen (INF-B-150, INF-B-180-CL, INF-B-180-MA, INF-B-180-STAT, INF-LGY, INF-LRS, MINF-B-180), Wahlpflichtmodul mit Pflichtveranstaltungen (INF-NF-30, INF-NF-60)
-----------------------	---

Verwendbarkeit	Dieses Modul wird in folgenden Studiengängen angeboten - INF-B-150: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 30 ECTS-Punkten - INF-B-180-CL: Bachelorstudiengang Informatik plus Computerlinguistik - INF-B-180-MA: Bachelorstudiengang Informatik plus Mathematik - INF-B-180-STAT: Bachelorstudiengang Informatik plus Statistik - INF-LGY: Lehramt Gymnasium - INF-LRS: Lehramt Realschule - INF-NF-30: Nebenfach: Informatik für Bachelorstudiengänge - INF-NF-60: Nebenfach: Informatik für Bachelorstudiengänge - MINF-B-180: Bachelorstudiengang Medieninformatik
-----------------------	---

Teilnahmevoraussetzungen	keine
---------------------------------	-------

Zeitpunkt im Studienverlauf	2. Semester (INF-B-180-STAT, INF-LGY, INF-NF-30, INF-NF-60, INF-LRS, INF-B-150, INF-B-180-CL, MINF-B-180), 4. Semester (INF-B-180-MA)
------------------------------------	---

Dauer	Das Modul erstreckt sich über 1 Semester.
--------------	---

Art der Bewertung	benotet
--------------------------	---------

Form der Modulprüfung	Klausur (90-180 Minute) oder mündlich (15-30 Minute) Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine GOP (Grundlagen und Orientierungsprüfung), auch für INF-NF-30, INF-NF-60
------------------------------	---

Modulverantwortlicher	Prof. Dr. Claudia Linnhoff-Popien
------------------------------	-----------------------------------

Anbieter Ludwig-Maximilians-Universität München
Fakultät für Mathematik, Informatik und Statistik
Institut für Informatik
Kerninformatik

**Unterrichts-
sprache(n)** Deutsch

Inhalte

Dieses Modul gibt einen Überblick über die binäre Darstellung von Informationen auf Computern, sowie über die Architektur und Arbeitsweise moderner Rechner nach von Neumann. Die klassischen Komponenten eines Computers werden eingeführt. Deren Interaktion wird zunächst theoretisch und dann mittels einer Maschinensprache und einer Assemblersprache praktisch behandelt. Es wird gezeigt, wie man mit Hilfe der Booleschen Algebra einfache Schaltungen und auch komplexere Komponenten eines Prozessors und des Speichers systematisch entwerfen und optimieren kann.

Im Einzelnen werden behandelt:

- Methoden zur binären Darstellung von Informationen im Rechner,
- Realisierung von Speicher durch Schaltwerke sowie durch optische und magnetische Medien,
- Boolesche Algebra zum Entwurf von Schaltungen,
- Entwurf und Optimierung einfacher logischer Schaltungen in Prozessoren,
- Komponenten der von Neumann Architektur und deren Optimierungen,
- maschinennahe Assemblerprogrammierung,
- das Zusammenspiel der unteren Ebenen eines Computers, sowie
- Parallelisierung und Mehrprozessorsysteme.

Literaturhinweise:

- Andrew S. Tanenbaum, Todd Austin, Rechnerarchitektur: Von der digitalen Logik zum Parallelrechner, 6. Auflage, ISBN-13: 978-3-86894-238-5,
- William Stallings, Computer Organization and Architecture: Designing for Performance, Pearson Education, 8th Edition, ISBN-13: 978-0135064177,
- David A. Patterson and John L. Hennessy, Morgan Kaufmann, Computer Organization and Design: The Hardware/Software Interface, 4th Edition, ISBN-13: 978-0123744937.

Das Modul besteht aus einer Vorlesung sowie Übungen in kleinen Gruppen. Die in der Vorlesung besprochenen Inhalte werden im Übungsteil anhand von praktischen Anwendungen eingeübt.

Qualifikationsziele

Die Studierenden entwickeln ein Grundverständnis des Entwurfs und der Architektur moderner Rechner und werden in den Zusammenhang zwischen höheren Programmiersprachen und der Abarbeitung einzelner Befehle auf Maschinenebene eingeführt. Insbesondere sollen sie ein Gefühl dafür entwickeln, welche Konsequenzen die Maschinenarchitektur für die Abarbeitung von Programmen hat, die in höheren Programmiersprachen geschrieben sind.

Die Studenten lernen sich schnell und umfangreich in komplexe Systeme und Zusammenhänge einzuarbeiten.

2.5 P 5: Softwareentwicklungspraktikum (INF-SEP)

Bemerkungen

Für die erfolgreiche Teilnahme am Praktikum sind Grundkenntnisse in der Programmiersprache Java unerlässlich. Anfang

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tu- rus	Präsenzzeit	Selbst- studium	ECTS
Tutorium	Softwareentwicklungspraktikum - Plenum	WiSe	30 h (2 SWS)	60 h	3 CP
Praktikum	Softwareentwicklungspraktikum - Praxis	WiSe	135 h (9 SWS)	135 h	9 CP

Im Modul müssen insgesamt 12 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 11 Wochenstunden. Inklusive Selbststudium sind etwa 360 Stunden aufzuwenden.

Art des Moduls	Pflichtmodul mit Pflichtveranstaltungen (INF-LGY, INF-LRS, INF-NF-60, MINF-NF-60), Wahlpflichtmodul mit Pflichtveranstaltungen (INF-B-120, INF-B-150, INF-B-180-CL, INF-B-180-MA, INF-B-180-STAT, MINF-B-180)
-----------------------	---

Verwendbarkeit	<p>Dieses Modul wird in folgenden Studiengängen angeboten</p> <ul style="list-style-type: none"> - INF-B-120: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 60 ECTS-Punkten - INF-B-150: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 30 ECTS-Punkten - INF-B-180-CL: Bachelorstudiengang Informatik plus Computerlinguistik - INF-B-180-MA: Bachelorstudiengang Informatik plus Mathematik - INF-B-180-STAT: Bachelorstudiengang Informatik plus Statistik - INF-LGY: Lehramt Gymnasium - INF-LRS: Lehramt Realschule - INF-NF-60: Nebenfach: Informatik für Bachelorstudiengänge - MINF-B-180: Bachelorstudiengang Medieninformatik - MINF-NF-60: Medieninformatik als Nebenfach für Bachelor und Masterstudiengänge
-----------------------	---

Teilnahmevoraussetzungen	keine
---------------------------------	-------

Zeitpunkt im Studienverlauf	3. Semester
------------------------------------	-------------

Dauer	Das Modul erstreckt sich über 1 Semester.
--------------	---

Art der Bewertung	benotet
--------------------------	---------

Form der mündlich (15-30 Minute)
Modulprüfung Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine

**Modul-
verantwort-
licher** Studiengangskoordinator(INF-LGY)

Anbieter Ludwig-Maximilians-Universität München
Fakultät für Mathematik, Informatik und Statistik
Institut für Informatik
Kerninformatik

**Unterrichts-
sprache(n)** Deutsch

Inhalte

Das Praktikum besteht aus einer Einführungsphase, in der grundlegende Programmier-techniken, die für die spätere Arbeit benötigt werden, eingeführt werden. Danach wird eine komplexe Softwareentwicklungsaufgabe in Teams von drei bis sechs Studierenden bearbeitet. Schwerpunkte des Praktikums liegen in der Erfahrung einer teamorientierten Softwareentwicklung unter Benutzung marktüblicher Werkzeuge und Methoden.

Die praktische Arbeit wird begleitet von einem Plenum, in dem auf die für das Praktikum erforderlichen Softwareentwicklungstechniken eingegangen wird. Dies umfasst typischerweise Programmieren mit Programmbibliotheken, Grafik-Programmierung, Aspekte der objektorientierte Analyse, Einführung in die Client-Server Programmierung und Verwendung von Software-Management-Tools. Außerdem werden auftretende aktuelle Probleme, Fragen und Schwierigkeiten bei der Software-Entwicklung diskutiert.

Die Studierenden arbeiten weitgehend selbständig in kleinen Teams. Jedem Team ist ein Betreuer zugeordnet, der dem Team bei den anstehenden Aufgaben hilft.

Qualifikationsziele

Das Softwareentwicklungspraktikum vermittelt praktische Erfahrung in der teamorientierten Entwicklung eines größeren und komplexen Software-Systems unter Benutzung marktüblicher Werkzeuge und Methoden. Es soll die Fähigkeit entwickelt werden, in einem kleinen Team ein größeres Softwareprojekt erfolgreich durchzuführen. Nach erfolgreicher Teilnahme am Softwareentwicklungspraktikum sollen sich die Teilnehmer trauen, Werkstudentenjobs in der IT-Industrie anzunehmen.

Bemerkungen

Für die erfolgreiche Teilnahme am Praktikum sind Grundkenntnisse in der Programmiersprache Java unerlässlich.

2.6 P 6: Formale Sprachen und Komplexität (INF-FSK)

Bemerkungen

Das Modul bildet die Grundvoraussetzung für darauf aufbauende Module aus dem Bereich der theoretischen Informatik und der formalen Softwaretechnik. Anfang

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tu- nus	Präsenzzeit	Selbst- studium	ECTS
Vorlesung	Vorlesung zu Formale Sprachen und Komplexität	SoSe	45 h (3 SWS)	45 h	3 CP
Übung	Übungen zu Formale Sprachen und Komplexität	SoSe	30 h (2 SWS)	60 h	3 CP

Im Modul müssen insgesamt 6 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 5 Wochenstunden. Inklusiv Selbststudium sind etwa 180 Stunden aufzuwenden.

Art des Moduls	Pflichtmodul mit Pflichtveranstaltungen (INF-B-120, INF-B-150, INF-B-180-CL, INF-B-180-MA, INF-B-180-STAT, INF-LGY, INF-LRS), Wahlpflichtmodul mit Pflichtveranstaltungen (INF-NF-30, INF-NF-60)
-----------------------	--

Verwendbarkeit	Dieses Modul wird in folgenden Studiengängen angeboten - INF-B-120: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 60 ECTS-Punkten - INF-B-150: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 30 ECTS-Punkten - INF-B-180-CL: Bachelorstudiengang Informatik plus Computerlinguistik - INF-B-180-MA: Bachelorstudiengang Informatik plus Mathematik - INF-B-180-STAT: Bachelorstudiengang Informatik plus Statistik - INF-LGY: Lehramt Gymnasium - INF-LRS: Lehramt Realschule - INF-NF-30: Nebenfach: Informatik für Bachelorstudiengänge - INF-NF-60: Nebenfach: Informatik für Bachelorstudiengänge
-----------------------	--

Teilnahmevoraussetzungen	keine
---------------------------------	-------

Zeitpunkt im Studienverlauf	2. Semester (INF-B-120), 4. Semester (INF-B-180-STAT, INF-LGY, INF-NF-30, INF-NF-60, INF-B-180-MA, INF-LRS, INF-B-150, INF-B-180-CL)
------------------------------------	--

Dauer	Das Modul erstreckt sich über 1 Semester.
--------------	---

Art der Bewertung	benotet
--------------------------	---------

Form der Klausur (90-180 Minute) oder mündlich (15-30 Minute)
Modulprüfung Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine

Modulverantwortlicher Prof. Dr. Hans Jürgen Ohlbach

Anbieter Ludwig-Maximilians-Universität München
Fakultät für Mathematik, Informatik und Statistik
Institut für Informatik
Kerninformatik

Unterrichtssprache(n) Deutsch

Inhalte

Das Modul vermittelt Grundkenntnisse in den Gebieten Formale Sprachen, Berechenbarkeit und Komplexitätstheorie.

Im einzelnen werden vermittelt:

- Automatentheorie und Formale Sprachen: Chomsky-Hierarchie, reguläre Sprachen und endliche Automaten, kontextfreie Sprachen und Kellerautomaten, kontextsensitive Sprachen,
- Berechenbarkeit: Turingmaschinen und andere Berechnungsmodelle, Unentscheidbarkeit, Halteproblem, rekursiv aufzählbare Probleme,
- Komplexitätstheorie, insbesondere die Klassen P und NP, Definition und Beweise für NP Vollständigkeit, Beispiele NP-vollständiger Probleme.

Literaturhinweise

- Theoretische Informatik kurzgefasst, Uwe Schöning, Spektrum Hochschultaschenbuch, ISBN 978-3-8274-1824-1

Das Modul besteht aus einer Vorlesung sowie Übungen in kleinen Gruppen. Die in der Vorlesung besprochenen Inhalte werden im Übungsteil anhand von praktischen Anwendungen eingeübt.

Vorkenntnisse

Grundkenntnisse aus den Mathematikvorlesungen.

Qualifikationsziele

Die Studierenden sollen die oben genannten immer wiederkehrenden theoretischen Grundlagen der Informatik kennenlernen und in die Lage versetzt werden, sie auf praktische Fragestellungen anzuwenden. Beispiele sind, ein vorgelegtes Problem als NP-vollständig zu identifizieren, oder zustandsorientierte Spezifikationen als endliche Automaten zu erkennen und Methoden wie Determinisierung und Minimierung darauf anzuwenden.

Darüber hinaus vertiefen die Studierenden ihre Fähigkeiten, abstrakte theoretische Inhalte zu verstehen, und mathematische Beweise nachzuvollziehen.

Bemerkungen

Das Modul bildet die Grundvoraussetzung für darauf aufbauende Module aus dem Bereich der theoretischen Informatik und der formalen Softwaretechnik.

2.7 P 7: Algorithmen und Datenstrukturen (INF-AIDs)

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tu- rus	Präsenzzeit	Selbst- studium	ECTS
Vorlesung	Vorlesung zu Algorithmen und Datenstrukturen	SoSe	45 h (3 SWS)	45 h	3 CP
Übung	Übungen zur Algorithmen und Datenstrukturen	SoSe	30 h (2 SWS)	60 h	3 CP

Im Modul müssen insgesamt 6 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 5 Wochenstunden. Inklusive Selbststudium sind etwa 180 Stunden aufzuwenden.

Art des Moduls	Pflichtmodul mit Pflichtveranstaltungen (INF-B-120, INF-B-150, INF-B-180-CL, INF-B-180-MA, INF-B-180-STAT, INF-LGY, INF-LRS, MINF-B-180), Wahlpflichtmodul mit Pflichtveranstaltungen (INF-NF-30, INF-NF-60)
-----------------------	--

Verwendbarkeit	<p>Dieses Modul wird in folgenden Studiengängen angeboten</p> <ul style="list-style-type: none"> - INF-B-120: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 60 ECTS-Punkten - INF-B-150: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 30 ECTS-Punkten - INF-B-180-CL: Bachelorstudiengang Informatik plus Computerlinguistik - INF-B-180-MA: Bachelorstudiengang Informatik plus Mathematik - INF-B-180-STAT: Bachelorstudiengang Informatik plus Statistik - INF-LGY: Lehramt Gymnasium - INF-LRS: Lehramt Realschule - INF-NF-30: Nebenfach: Informatik für Bachelorstudiengänge - INF-NF-60: Nebenfach: Informatik für Bachelorstudiengänge - MINF-B-180: Bachelorstudiengang Medieninformatik
-----------------------	--

Teilnahmevoraussetzungen	keine
---------------------------------	-------

Zeitpunkt im Studienverlauf	2. Semester (INF-B-120, INF-NF-60, INF-LRS, INF-B-150, INF-B-180-CL, MINF-B-180), 4. Semester (INF-B-180-STAT, INF-LGY, INF-NF-30, INF-B-180-MA)
------------------------------------	--

Dauer	Das Modul erstreckt sich über 1 Semester.
--------------	---

Art der Bewertung	benotet
--------------------------	---------

Form der Modulprüfung Klausur (90-180 Minute) oder mündlich (15-30 Minute)
Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine
GOP (Grundlagen und Orientierungsprüfung), auch für INF-B-120, INF-NF-60

Modulverantwortlicher Dr. Matthias Schubert

Anbieter Ludwig-Maximilians-Universität München
Fakultät für Mathematik, Informatik und Statistik
Institut für Informatik
Kerninformatik

Unterrichtssprache(n) Deutsch

Inhalte

Dieses Modul gibt eine Einführung in die Entwicklung effizienter Algorithmen sowie das Zusammenspiel zwischen Algorithmus und Datenstruktur.

Grundbegriffe zu Algorithmen und Laufzeitanalyse

- Abgrenzung verschiedener Laufzeitabschätzungen (best-case, worst-case, erwartete Laufzeitkomplexität)
- Asymptotische Analyse von oberen und unteren Schranken der Laufzeitkomplexität
- Groß O Notation (Definition und Berechnung)
- Wichtige Komplexitätsklassen (konstant, logarithmisch, linear, quadratisch, und exponentiell)
- Methoden der empirischen Performanz Evaluation
- Trade-Off zwischen Zeit- und Speicherverbrauch von Algorithmen.

Optimaler Inhalte:

- klein o, groß Omega und Groß Theta Notation
- rekurrente Relationen
- Analyse von iterativen und rekursiven Algorithmen
- Hauptsatz der Laufzeitfunktionen.

Grundlegende Datenstrukturen und Algorithmen

- Elementare Datentypen (Integer, Float, Strings etc.)
- Verbunddatentypen, Objekte und Arrays
- dynamische Datenstrukturen (einfach und doppelt verkettete Listen, Stapel, Warteschlangen, Bäume)
- Implementierung von dynamischen Datenstrukturen

- einfache numerische Algorithmen (z.B. Bestimmen des Durchschnittswerts, Maximum, oder Minimum einer Liste oder eines Arrays, approximative Berechnung der Quadratwurzel, Bestimmung des größten gemeinsamen Teilers)
- sequentielle und binäre Suche in Arrays.

Datenstrukturen und Algorithmen zur Schlüsselsuche

- Zusammenhang Suchzeit, Einfügezeit, Löschenzeiten, und Speicherplatzbedarf.
- Balancierte Suchbäume (Grundprinzip und Analyse, Beipielalgorithmen z.B. AVL-Bäume, Rot-Schwarzbäume Bäume)
- SuchBäume für den Sekundärspeicher (Problemstellung, z.B. B-Bäume)
- Grundprinzip des Hashing (Einfache Hashfunktionen, Kollisionsstrategien)
- dynamische Hash-Verfahren(z.B. lineares Hashing).

Optionale Inhalte:

- weiterführende Algorithmen zur Schlüsselsuche im Hauptspeicher (z.B. optimale binäre Suchbäume, Splay trees, Treaps)
- weiterführende Indexstrukturen für den sekundärspeicher(z.B. B*-Baum)
- weiterführende Hash-Verfahren für den Sekundärspeicher (z.B. linearen Hashing mit partiellen Erweiterungen).

Sortierverfahren

- elementare Sortieralgorithmen (Sortieren durch Abzählen, Insertion Sort, Selection Sort, BubbleSort)
- fortgeschrittene Sortierverfahren (Heapsort, Quicksort)
- Sortieren auf dem Sekundärspeicher (Merge-Sort)
- untere Schranke für vergleichsbasiertes Sortieren
- Schlüsselbasiertes Sortieren (Bucketsort).

Optionale Inhalte

- weiterführende Verfahren für das Sortieren großer Schlüsselmengen (button-up Heapsort, clever Quicksort)
- weiterführenden Verfahren zur Schlüsselbasierten Suche (Radix-Sort)
- Priortätswarteschlagen (z.B. Fibonacci-Heaps).

Graphalgorithmen

- Grundlegende Eigenschaften von Graphen
- Darstellung von Graphen (Adjanzenzmatrix, Adjanzenzlisten)
- Graphdurchläufe (Breitendurchlauf, Tiefendurchlauf)
- Bestimmung kürzester Pfade (Dijkstra's und Floyd's algorithms)
- Minimale Spannbäume (Algorithmen von Prim und Kruskal).

Optionale Inhalte

- Flüsse in Netzwerken (z.B. maximaler Fluß, Max-Flow–Min-Cut Theorem, Maximales bipartites Matching)
- weitere Graphprobleme(z.B. Topologische Sortierung, Finden stark verbundenen Komponenten, Graph Matching).

Algorithmische Strategien

- Erschöpfende Suche
- Greedy Algorithmen
- Divide-and-conquer
- Rekursives Backtracking
- Branch-and-bound.

Optinale Inhalte

- Reduktion: Transform-and-Conquer.

Optionale Kapitel

- Lineare Programmierung (Duality, Simplex Algorithmen, Innere Punkt Verfahren)
- Pattern matching und String/Text Algorithmen (z.B. Substring Matching, Reguläre Ausdrücke, Längste gemeinsame Teilsequenzen)
- String-basierte Datenstrukturen und Algorithmen (z.B. Suffix Arrays, Suffix Trees, Tries)
- Algorithmen zur Lösung numerischer Probleme (z.B. Primzahlentests, Integer Faktorisierung)
- Geometrische Datenstrukturen und Algorithmen (z.B.Darstellung von Punkten, Liniensegmenten und Polygonen, Eigenschaften und Schnittpunkte, konvexe Hüllen, räumliche Zerlegung, Kollisionserkennung, geometrische Nähe).

Literatur

- R. Sedgewick: Algorithmen in Java, 2. Auflage, Pearson Studium,
- T. Ottmann, P. Widmayer: Algorithmen und Datenstrukturen, 4. Auflage, Spektrum Akademischer Verlag, 2002
- T. H. Cormen, C. Leiserson, R. Rivest, C. Stein: Algorithmen - Eine Einführung,4. Auflage Oldenbourg, 2013

Das Modul besteht aus einer Vorlesung sowie Übungen in kleinen Gruppen. Die in der Vorlesung besprochenen Inhalte werden im Übungsteil anhand von praktischen Anwendungen eingeübt.

Qualifikationsziele

Bei erfolgreichem Abschluss des Moduls sollten die Teilnehmer(innen) Kenntnisse über die folgenden Themen erlangt haben:

- Grundlegende Eigenschaften und Entwurfmethoden für Algorithmen
- Effiziente Algorithmen und Datenstrukturen für grundlegende Probleme

- Wesentliche Komplexitätsklassen für das Laufzeitverhalten und den Speicherplatzbedarf von Algorithmen.

Teilnehmer(innen) des Moduls erlernen die Fähigkeit zur:

- Analyse des Laufzeitverhaltens und des Speicherplatzbedarf für gegebene Algorithmen.
- Formalen Modellierung von algorithmen Problemstellungen
- Adaption bekannter Datenstrukturen und Algorithmen an modifizierte Problemstellungen.

Unter Anwendung der erlangten Fähigkeiten und Kenntnisse sollen die Teilnehmer in die Lage versetzt werden:

- Programme unter Verwendung der erlernten algorithmischen Techniken eigenständig zu konzipieren und in einer Programmiersprache zu implementieren.
- Für ein gegebenes Problem die Verwendung verschiedener Lösungsmöglichkeiten nach formalen Kriterien zu beurteilen.

2.8 P 8: Datenbanksysteme I (INF-DBSI)

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tu- nus	Präsenzzeit	Selbst- studium	ECTS
Vorlesung	Vorlesung zu Datenbanksysteme I	WiSe	30 h (2 SWS)	60 h	3 CP
Übung	Übungen zu Datenbanksysteme I	WiSe	30-45 h (2-3 SWS)	45 h - 60 h	3 CP

Im Modul müssen insgesamt 6 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 5 Wochenstunden. Inklusive Selbststudium sind etwa 180 Stunden aufzuwenden.

Art des Moduls	Pflichtmodul mit Pflichtveranstaltungen (INF-B-120, INF-B-150, INF-B-180-CL, INF-B-180-MA, INF-B-180-STAT, INF-LGY, INF-LRS, MINF-B-180), Wahlpflichtmodul mit Pflichtveranstaltungen (INF-NF-30, INF-NF-60)
-----------------------	--

Verwendbarkeit	Dieses Modul wird in folgenden Studiengängen angeboten - INF-B-120: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 60 ECTS-Punkten - INF-B-150: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 30 ECTS-Punkten - INF-B-180-CL: Bachelorstudiengang Informatik plus Computerlinguistik - INF-B-180-MA: Bachelorstudiengang Informatik plus Mathematik - INF-B-180-STAT: Bachelorstudiengang Informatik plus Statistik - INF-LGY: Lehramt Gymnasium - INF-LRS: Lehramt Realschule - INF-NF-30: Nebenfach: Informatik für Bachelorstudiengänge - INF-NF-60: Nebenfach: Informatik für Bachelorstudiengänge - MINF-B-180: Bachelorstudiengang Medieninformatik
-----------------------	--

Teilnahmevoraussetzungen	keine
---------------------------------	-------

Zeitpunkt im Studienverlauf	5. Semester
------------------------------------	-------------

Dauer	Das Modul erstreckt sich über 1 Semester.
--------------	---

Art der Bewertung	benotet
--------------------------	---------

Form der Modulprüfung	Klausur (90-180 Minute) oder mündlich (15-30 Minute) Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine
------------------------------	--

Modul- verantwort- licher	Prof. Dr. Christian Böhm
--	--------------------------

Anbieter	Ludwig-Maximilians-Universität München Fakultät für Mathematik, Informatik und Statistik Institut für Informatik Kerninformatik LFE Datenbanksysteme
-----------------	--

Unterrichts- sprache(n)	Deutsch
------------------------------------	---------

Inhalte

Die Vorlesung bietet eine Einführung in das Gebiet der Datenbanksysteme aus Anwendersicht. Im Mittelpunkt stehen die theoretischen Aspekte des relationalen Datenbankentwurfs anhand des relationalen Datenmodells, der relationalen Algebra und des Relationenkalküls. Es erfolgt eine ausführliche Behandlung der Anfragesprache SQL, die in den meisten relationalen Systemen implementiert ist. Des Weiteren werden Formalismen, Theorie und Algorithmen der relationalen Entwurfstheorie beschrieben und neuere Anwendungen im Bereich Datenbanken behandelt.

Im einzelnen werden vermittelt:

- Relationales und objektrelationales sowie weitere Datenmodelle,
- Relationale Algebra,
- Tupel- und Bereichskalkül,
- SQL,
- Datenbankentwurf nach dem E/R-Modell,
- Normalformen,
- Transaktionen incl. Synchronisations- und Recovery-Techniken,
- Physischer Datenbankentwurf (Indexstrukturen und Queryoptimierung),
- Integration von Datenbankoperationen in Anwendungsprogramme.

Das Modul besteht aus einer Vorlesung mit Übungen in kleinen Gruppen. Die in der Vorlesung besprochenen Inhalte werden im Übungsteil anhand von praktischen Anwendungen eingeübt.

Qualifikationsziele

Die Studierenden werden in die Lage versetzt, Datenbanksysteme als Anwender, als Anwendungsprogrammierer sowie als Systemdesigner professionell anzuwenden. Es wird die Kompetenz vermittelt, mittels komplexer Anfragen in umfangreichen Datenbanken zielgerichtet zu recherchieren, Datenbank-Schemata unter Vermeidung von Redundanzproblemen und unter Berücksichtigung von Effizienzaspekten systematisch zu entwerfen, und effiziente Datenbank-Anwendungen zu implementieren.

2.9 P 9: Softwaretechnik (INF-SWT)

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tur- nus	Präsenzzeit	Selbst- studium	ECTS
Vorlesung	Vorlesung zu Softwaretechnik	WiSe	45 h (3 SWS)	45 h	3 CP
Übung	Übungen zu Softwaretechnik	WiSe	30 h (2 SWS)	60 h	3 CP

Im Modul müssen insgesamt 6 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 5 Wochenstunden. Inklusive Selbststudium sind etwa 180 Stunden aufzuwenden.

Art des Moduls	Wahlpflichtmodul mit Pflichtveranstaltungen (INF-LGY, INF-LRS, INF-NF-30, INF-NF-60), Pflichtmodul mit Pflichtveranstaltungen (INF-B-120, INF-B-150, INF-B-180-CL, INF-B-180-MA, INF-B-180-STAT, MINF-B-180)
-----------------------	--

Verwendbarkeit	Dieses Modul wird in folgenden Studiengängen angeboten - INF-B-120: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 60 ECTS-Punkten - INF-B-150: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 30 ECTS-Punkten - INF-B-180-CL: Bachelorstudiengang Informatik plus Computerlinguistik - INF-B-180-MA: Bachelorstudiengang Informatik plus Mathematik - INF-B-180-STAT: Bachelorstudiengang Informatik plus Statistik - INF-LGY: Lehramt Gymnasium - INF-LRS: Lehramt Realschule - INF-NF-30: Nebenfach: Informatik für Bachelorstudiengänge - INF-NF-60: Nebenfach: Informatik für Bachelorstudiengänge - MINF-B-180: Bachelorstudiengang Medieninformatik
-----------------------	--

Teilnahmevoraussetzungen	keine
---------------------------------	-------

Zeitpunkt im Studienverlauf	3. Semester (INF-B-180-STAT, INF-B-120, INF-NF-30, INF-B-150, INF-B-180-CL), 5. Semester (INF-LGY, INF-NF-60, INF-B-180-MA, MINF-B-180), 7. Semester (INF-LRS)
------------------------------------	--

Dauer	Das Modul erstreckt sich über 1 Semester.
--------------	---

Art der Bewertung	benotet
--------------------------	---------

Form der Modulprüfung	Klausur (90-180 Minute) oder mündlich (15-30 Minute) Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine
------------------------------	--

Modul- verantwort- licher	Prof. Dr. Rolf Hennicker
--	--------------------------

Anbieter	Ludwig-Maximilians-Universität München Fakultät für Mathematik, Informatik und Statistik Institut für Informatik Kerninformatik LFE Programmierung und Softwaretechnik
-----------------	--

Unterrichts- sprache(n)	Deutsch, Englisch
------------------------------------	-------------------

Inhalte

In diesem Modul werden die wesentlichen Prinzipien der Softwaretechnik besprochen. Der gesamte Softwareentwicklungsprozess wird vorgestellt, beginnend mit der Anforderungsanalyse über den Systementwurf bis zu Implementierung und Test. Als grafische Modellierungssprache wird die Unified Modeling Language (UML) im gesamten Entwicklungsprozess eingesetzt; als Implementierungssprache wird Java verwendet.

Die inhaltlichen Schwerpunkte des Moduls sind:

- Softwareentwicklungsprozesse,
- Anforderungsanalyse unter Verwendung von Anwendungsfällen,
- Entwurf von statischen Systemstrukturen mit Klassendiagrammen,
- Verhaltensmodellierung mit Zustands-, Sequenz- und Aktivitätsdiagrammen,
- Architektur komplexer Softwaresysteme,
- Design- und Architekturmuster,
- Beziehung zwischen Modellen und Implementierungen in objektorientierten Sprachen,
- Testen von Software.

Der Modul besteht aus einer Vorlesung und Übungen in Gruppen. Die in der Vorlesung besprochenen Inhalte werden in den Übungen anhand von praktischen Anwendungsbeispielen eingeübt.

Qualifikationsziele

Die Studierenden erwerben ein allgemeines Verständnis der wichtigsten Aspekte moderner Softwareentwicklung unter Anwendung von Notationen und Werkzeugen gemäß des Stands der Forschung und der industriellen Praxis. Sie sind in der Lage strukturelle und dynamische Eigenschaften komplexer Softwaresysteme zu modellieren und die Modelle in Software zu überführen.

2.10 P 10: Vertiefende Themen aus der theoretischen Informatik (INF-VT)

Zugeordnete Module:

Lehrform	Veranstaltung	Tur- nus	Präsenzzeit	Selbst- studium	ECTS
Vorlesung	Vorlesung zu Vertiefende Themen aus der theoretischen Informatik	SoSe	45 h (3 SWS)	60 h	4 CP
Übung	Übungen zu Vertiefende Themen aus der theoretischen Informatik	SoSe	30 h (2 SWS)	45 h	2 CP

Im Modul müssen insgesamt 6 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 5 Wochenstunden. Inclusive Selbststudium sind etwa 180 Stunden aufzuwenden.

2.11 P 11: Vertiefende Themen aus der technischen Informatik (INF-VT)

Zugeordnete Module:

Lehrform	Veranstaltung	Tur- nus	Präsenzzeit	Selbst- studium	ECTS
Vorlesung	Vorlesung zu Vertiefende Themen aus der technischen Informatik	SoSe	45 h (3 SWS)	60 h	4 CP
Übung	Übungen zu Vertiefende Themen aus der technischen Informatik	SoSe	30 h (2 SWS)	45 h	2 CP

Im Modul müssen insgesamt 6 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 5 Wochenstunden. Inclusive Selbststudium sind etwa 180 Stunden aufzuwenden.

2.12 P 12: Informatikseminar (INF-Sem)

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tur- nus	Präsenzzeit	Selbst- studium	ECTS
Seminar	Seminar zu ausgewählten Themen der Informatik	WiSe, SoSe	30 h (2 SWS)	60 h	3 CP

Im Modul müssen insgesamt 3 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 2 Wochenstunden. Inklusive Selbststudium sind etwa 90 Stunden aufzuwenden.

Art des Moduls Pflichtmodul mit Pflichtveranstaltungen

Verwendbarkeit Dieses Modul wird in folgenden Studiengängen angeboten

- INF-B-120: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 60 ECTS-Punkten
- INF-B-150: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 30 ECTS-Punkten
- INF-B-180-CL: Bachelorstudiengang Informatik plus Computerlinguistik
- INF-B-180-MA: Bachelorstudiengang Informatik plus Mathematik
- INF-B-180-STAT: Bachelorstudiengang Informatik plus Statistik
- INF-LGY: Lehramt Gymnasium
- INF-LRS: Lehramt Realschule

Teilnahmevoraussetzungen keine

Zeitpunkt im Studienverlauf 2. Semester (INF-B-180-MA), 3. Semester (INF-B-180-CL), 4. Semester (INF-B-180-STAT, INF-B-120, INF-LRS, INF-B-150), 7. Semester (INF-LGY)

Dauer Das Modul erstreckt sich über 1 Semester.

Art der Bewertung benotet

Form der Modulprüfung Seminar (7000-14000 Zeichen)
Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine

Modulverantwortlicher Studiengangskoordinator(INF-LGY)

Anbieter Ludwig-Maximilians-Universität München
Fakultät für Mathematik, Informatik und Statistik
Institut für Informatik

**Unterrichts-
sprache(n)** Deutsch

Inhalte

Das Seminar behandelt aktuelle Themen aus den Forschungsschwerpunkten der Informatik.

Es werden individuelle Themen der Informatik an ein bis zwei Studierende vergeben. Die Studierenden müssen sich in diese Themen einarbeiten, selbständig eine Hausarbeit anfertigen und einen Vortrag vorbereiten. Sie tragen diesen Vortrag im Seminar vor und stellen sich einer kritischen Diskussion.

Qualifikationsziele

In dem Seminar werden die selbständige Erarbeitung eines komplizierten Themas sowie Präsentations- und Vortragstechniken eingeübt.

2.13 P 13.1: Grundlagen der Fachdidaktik II (INF-FD2-GY)

Zuordnung zum Studiengang: Lehramt Gymnasium (72 CP)

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tur- nus	Präsenzzeit	Selbst- studium	ECTS
Vorlesung	Grundlagen der Fachdidaktik II	SoSe	30 h (2 SWS)	90 h	3 CP

Im Modul müssen insgesamt 3 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 2 Wochenstunden. Inklusive Selbststudium sind etwa 90 Stunden aufzuwenden.

Art des Moduls Pflichtmodul mit Pflichtveranstaltungen

Teilnahmevoraussetzungen keine

Zeitpunkt im Studienverlauf 5./6. Semester

Dauer Das Modul erstreckt sich über 1 Semester.

Art der Bewertung benotet

Form der Modulprüfung Klausur (60-120 Minute) oder mündlich (20-40 Minute) oder Hausarbeit (30-40 Seiten)
Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine

Modulverantwortlicher Prof. PhD Martin Hofmann

Anbieter Ludwig-Maximilians-Universität München
Fakultät für Mathematik, Informatik und Statistik
Institut für Informatik

Unterrichtssprache(n) Deutsch

Inhalte

Dieses Modul gibt eine Einführung in die Methodik und Didaktik des Informatikunterrichts am Gymnasium in Bayern. Neben allgemeinen Konzepten sind konkrete Umsetzungsmöglichkeiten

für einzelne Lehrplaninhalte und Unterrichtssituationen auch ein Schwerpunkt.

Im Einzelnen werden vermittelt:

- Überblick über die Unterrichtsbedingungen am Gymnasium,
- Vorstellung des Fachprofils und der Lehrpläne der verschiedenen Jahrgangsstufen,
- Beitrag der Informatik zum Bildungs- und Erziehungsauftrag,
- Konkrete Umsetzungsmöglichkeiten einzelner Lehrplaninhalte,
- Auswahl und Schwerpunktssetzung von Unterrichtsinhalten,
- Konzepte und Theorien aus der Fachdidaktik Informatik,
- methodische Ansätze im Informatikunterricht,
- Bedeutung und Einsatzmöglichkeiten für den Informatikunterricht von beispielsweise Projekten, Lernplattformen, Aufgabenkonstruktion und Binnendifferenzierung.

Das Modul besteht aus einer Vorlesung, in der Diskussionen und praktische Übungen einen festen Bestandteil haben.

Qualifikationsziele

Die Studierenden kennen die Lehrplaninhalte am Gymnasium und wissen, welchen Beitrag die Informatik zum Erziehungs- und Bildungsauftrag leistet. Sie können unter Berücksichtigung didaktischer und methodischer Konzepte Informatikunterricht planen.

2.14 P 13.2 und 13.3: Fachdidaktisches Praktikum und Seminar (INF-FDPS-GY)

Zuordnung zum Studiengang: Lehramt Gymnasium (72 CP)

Bemerkung:

Die beiden Module *Grundlagen der Fachdidaktik II* (INF-FD2) und *Fachdidaktisches Praktikum und Seminar* (INF-FDPS) sind in der Prüfungsordnung zu einem Modul *Fachdidaktik II* zusammengefasst.

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tur- nus	Präsenzzeit	Selbst- studium	ECTS
Praktikum	Fachdidaktisches Praktikum	SoSe	45 h (3 SWS)	45 h	3 CP
Seminar	Fachdidaktisches Seminar	SoSe	30 h (2 SWS)	60 h	3 CP

Im Modul müssen insgesamt 6 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 5 Wochenstunden. Inklusive Selbststudium sind etwa 180 Stunden aufzuwenden.

Art des Moduls Pflichtmodul mit Pflichtveranstaltungen

Teilnahmevoraussetzungen keine

Zeitpunkt im Studienverlauf 7./8. Semester

Dauer Das Modul erstreckt sich über ein Semester.

Art der Bewertung

Form der Modulprüfung Klausur (60-75 min), mündliche Prüfung (20-30 min) oder Hausarbeit (30-40 Seiten)
Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine

Modulverantwortlicher Prof. PhD Martin Hofmann

Anbieter Ludwig-Maximilians-Universität München
Fakultät für Mathematik, Informatik und Statistik
Institut für Informatik

Unterrichtssprache(n) Deutsch

Inhalte

Im Praktikum erhalten die Studierenden Einblick in verschiedene für den Schulunterricht mögliche Werkzeuge und Unterrichtsmethoden. Insbesondere sind dies:

- schülernahe Entwicklungsumgebungen,
- Projektmanagement an Schulen,
- Softwaresysteme für den Informatikunterricht (didaktische Tools),
- Durchführung eines Projekts mit einer Schülergruppe,
- pädagogische Benutzeroberflächen für Schulnetzwerke.

Im Seminar erarbeiten und diskutieren die Studierenden auf der Grundlage ihrer fachwissenschaftlichen und fachdidaktischen Vorkenntnisse Unterrichtsentwürfe für einzelne ausgewählte Sequenzen aus den aktuellen Lehrplänen. Dabei soll das gesamte Spektrum der gymnasialen Informatik von der Unter- bis zur Oberstufe, zu den Seminaren und zum Wahlunterricht abgedeckt werden. Daher ist es nützlich, wenn man vor dem Seminar bereits beide Vorlesungen Fachdidaktik I und Fachdidaktik II erfolgreich besucht hat

Qualifikationsziele

Ziel des Praktikums ist es, einen Überblick über geeignete schülernahe Softwarewerkzeuge zu erhalten und ihre Einsatzmöglichkeiten bewerten zu können.

Ziel des Seminars ist, die fachwissenschaftlichen und fachdidaktischen Inhalte des Studiums auf den konkreten, aktuellen Lehrplan anwenden zu können. Die Studierenden werden dadurch mit den theoretischen Aspekten der Unterrichtsvorbereitung vertraut.

2.15 P 14: Vertiefende Themen aus Datenbanksysteme und Softwaretechnologie (INF-VT)

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tur- nus	Präsenzzeit	Selbst- studium	ECTS
Vorlesung	Vorlesung zu Vertiefende Themen aus Datenbanksysteme und Softwaretechnologie	SoSe	45 h (3 SWS)	60 h	4 CP
Übung	Übungen zu Vertiefende Themen aus Datenbanksysteme und Softwaretechnologie	SoSe	30 h (2 SWS)	45 h	2 CP

Im Modul müssen insgesamt 6 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 5 Wochenstunden. Inklusive Selbststudium sind etwa 180 Stunden aufzuwenden.

2.16 P 15: Praktische Programmierung (INF-PP)

Zuordnung zum Studiengang: Lehramt Gymnasium

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tur- nus	Präsenzzeit	Selbst- studium	ECTS
Praktikum Praktische Program- mierung		0 h	180 h	6 CP	

Im Modul müssen insgesamt 6 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 0 Wochenstunden. Inklusive Selbststudium sind etwa 180 Stunden aufzuwenden.

**Art des Mo-
duls** Pflichtmodul mit Pflichtveranstaltungen

**Teilnahme-
voraus-
setzungen** keine

**Zeitpunkt
im Studien-
verlauf** 8. Semester (INF-GY)

Dauer Das Modul erstreckt sich über 1 Semester.

**Art der Be-
wertung** benotet

**Form der
Modulprüfung** Hausarbeit (30-50 Seiten) und mündlich (30 Minute)
Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine

**Modul-
verantwort-
licher** Prof. PhD Martin Hofmann

Anbieter Ludwig-Maximilians-Universität München
Fakultät für Mathematik, Informatik und Statistik
Institut für Informatik

**Unterrichts-
sprache(n)** Deutsch

Inhalte

Die Veranstaltung entspricht dem Praktikum zur planmäßigen Entwicklung eines größeren Softwaresystems nach LPO I, §69.

Das Thema des Programmierpraktikums ist frei wählbar. Dabei gibt es mehrere Möglichkeiten:

- das Thema wird durch einen Dozenten gestellt,
- das Thema wird durch einen Gymnasial- oder Realschullehrer (einem abgeordneten Lehrer am Lehrstuhl) gestellt,
- es wird ein Thema selbst gewählt,
- die Arbeit wird in oder mit einer Firma durchgeführt.

Zusätzlich zur Programmierarbeit muss eine ausführliche Dokumentation abgegeben werden.

Qualifikationsziele

Die Studierenden setzen die bisher erworbenen Kenntnisse über Softwareentwicklung selbstständig und in einem eigenen Projekt um. Dabei erfahren sie, wie planvolles Vorgehen, Anwendung von Entwurfsmustern und geeignete Testverfahren zum Erfolg führen.

3 Vertiefende Themen

Die folgenden Module sind eine Auswahl, die als Vertiefende Themen angerechnet werden können. Diese Module dienen der Verbreiterung und Vertiefung der Kenntnisse und Fähigkeiten. Etliche davon sind an aktuelle Schwerpunkte in der Forschung des Lehrpersonals angelehnt und dienen damit der konsequenten Umsetzung des Prinzips der Forschungsorientierung in der Lehre. Durch den Besuch dieser Veranstaltungen werden Studierende bereits frühzeitig an aktuelle Fragen der Forschung herangeführt und erhalten Einblick in die Weiterentwicklung des Fachs. Module aus den Masterstudiengängen können nur für Studierende empfohlen werden, die im bisherigen Bachelorstudiengang schon hervorragende Leistungen gezeigt haben.

3.1 VT 1: Mobilkommunikation (INF-MK)

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tu- rus	Präsenzzeit	Selbst- studium	ECTS
Vorlesung	Vorlesung Mobilkommunikation	WiSe	45 h (3 SWS)	75 h	4 CP
Übung	Übungen zu Mobilkommunikation	SoSe	15 h (1 SWS)	45 h	2 CP

Im Modul müssen insgesamt 6 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 4 Wochenstunden. Inklusive Selbststudium sind etwa 180 Stunden aufzuwenden.

Art des Moduls Wahlpflichtmodul mit Pflichtveranstaltungen

Verwendbarkeit Dieses Modul wird in folgenden Studiengängen angeboten

- INF-LGY: Lehramt Gymnasium
- INF-M-120: Masterstudiengang Informatik
- MINF-M-120: Masterstudiengang Medieninformatik

Zeitpunkt im Studienverlauf 1. Semester

Dauer Das Modul erstreckt sich über 1 Semester.

Art der Bewertung benotet

Form der Modulprüfung Klausur (90-180 Minute)
Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine

Modulverantwortlicher Prof. Dr. Claudia Linnhoff-Popien

Anbieter Ludwig-Maximilians-Universität München
Fakultät für Mathematik, Informatik und Statistik
Institut für Informatik
Kerninformatik
LFE Mobile und Verteilte Systeme

Unterrichtssprache(n) Deutsch

Inhalte

Die rasante und weltweite Verbreitung von 3G/4G-Mobilfunknetzen sowie die Verbreitung von WLAN-fähigen Endgeräten und Routern für den Heimbereich haben dazu geführt, dass man heutzutage fast überall kabellos ins Internet gelangen kann. Der Vision des Ubiquitous Computing entsprechend, ist das mobile Internet so zu einer für uns selbstverständlichen Technologie geworden. Die Vorlesung Mobilkommunikation bietet einen Überblick über die technischen Grundlagen und die konkrete Umsetzung von kabelloser Kommunikation in modernen Szenarien. Dabei behandelt sie neben den funktechnologischen Grundlagen auch die organisatorischen Aspekte größerer Mobilfunknetze.

Themen der Vorlesung sind u.a.:

- Mobilität und Mobilitätsformen,
- Grundlagen der Kommunikation (Shannon-Modell, Shannon-Theorem, Nyquist, ...),
- Antennen, Funkausbreitung und Funkübertragung (Friis-Modell, Motley-Keenan, Fading, Inverse Square Law, Gain, Apertur, ...),
- Modulationsverfahren (AM, FM, PM, BFSK, DTMF, BPSK, Diff-BPSK, QPSK, OQPSK, QAM, MSK, GMSK, DSSS, OFDM),
- Multiplexing-Verfahren (FDM, TDM, CDM, m-Sequences, Gold-Codes, OVSF, ODFMA),
- Mobilfunknetze (GSM, UMTS, LTE),
- lokale Funknetze (Bluetooth, NFC, 802.11),
- Mobilitätsmanagement (Channel Assignment Schemes, Mobile IP, ...),
- Planung von Mobilfunknetzen,
- Location-based Services.

Literaturhinweise:

- Stallings: Wireless Communications and Networks, ISBN: 978-8131709733
- Schiller: Mobilkommunikation, ISBN: 978-3827370600
- Küpper: Location-based Services, ISBN: 978-0470092316

Das Modul besteht aus einer Vorlesung sowie Übungen in kleinen Gruppen. Die in der Vorlesung besprochenen Inhalte werden im Übungsteil anhand von praktischen Anwendungen eingeübt.

Vorkenntnisse

Grundlagen der Analysis, Grundlagen von Rechnernetzen

Qualifikationsziele

Dieses Modul bietet Studenten eine tiefgehende Behandlung von Mobilfunknetzen, ausgehend von den Grundlagen der Funkübertragung über Medienzugriff und Multiplexing bis hin zu Netzplanung und Eigenschaften zellulärer Netze. Weiter werden historische und aktuelle Funknetze eingeführt und im Detail diskutiert. Ein Student ist in der Lage, den aktuellen Entwicklungen in der Mobilkommunikation zu folgen und die Lösungen und Ideen auf neue Situationen zu übertragen.

Die vermittelten Grundlagen der Funkkommunikation geben darüber hinaus Einblick in Bereiche der Physik und Elektrotechnik.

3.2 VT 2: Sensornetze (INF-SN)

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tur- nus	Präsenzzeit	Selbst- studium	ECTS
Vorlesung	Vorlesung Sensornetze	SoSe	45 h (3 SWS)	75 h	4 CP
Übung	Übungen zu Sensornetzen	SoSe	30 h (2 SWS)	30 h	2 CP

Im Modul müssen insgesamt 6 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 5 Wochenstunden. Inklusive Selbststudium sind etwa 180 Stunden aufzuwenden.

Art des Moduls Wahlpflichtmodul mit Pflichtveranstaltungen

Verwendbarkeit Dieses Modul wird in folgenden Studiengängen angeboten
- INF-LGY: Lehramt Gymnasium
- INF-M-120: Masterstudiengang Informatik
- MINF-M-120: Masterstudiengang Medieninformatik

Zeitpunkt im Studienverlauf 2. Semester

Dauer Das Modul erstreckt sich über 1 Semester.

Art der Bewertung benotet

Form der Modulprüfung Klausur (90-180 Minute) oder mündlich (15-30 Minute)
Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine

Modulverantwortlicher Prof. Dr. Claudia Linnhoff-Popien

Anbieter Ludwig-Maximilians-Universität München
Fakultät für Mathematik, Informatik und Statistik
Institut für Informatik
Kerninformatik
LFE Mobile und Verteilte Systeme

Unterrichtssprache(n) Deutsch

Inhalte

Das Modul bietet einen Überblick über Theorie und Anwendungen aus dem Bereich der Sensornetze. Sensornetze sind spezielle Computer-Netzwerke, bei denen eine Vielzahl von Knoten in kooperativer Weise Messdaten erfassen und verarbeiten müssen. Eine besondere Rolle nehmen dabei die ungewarteten Funk-Sensornetze ein. Bei diesen Sensornetzen muss die gesamte Architektur daran angepasst werden, dass die begrenzten Energie-Ressourcen der einzelnen Knoten zur Erfüllung der gemeinsamen Messaufgabe möglichst effizient eingesetzt werden können.

Themen sind u.a.:

- Architektur von Sensorknoten und -netzen,
- Spezielle Anforderungen für Sensornetze,
- Elemente der Funkübertragung,
- Medienzugriffsverfahren,
- Fehlerkontrolle und -korrektur,
- Routing und Adressierung,
- Topologieerkennung und -kontrolle,
- Zeitsynchronisation,
- Lokalisierung in Sensornetzen
- Beispiele.

Literaturhinweise:

- Dargie, Poellabauer: Fundamentals of Wireless Sensor Networks: Theory and Practice,
- Akyildiz, Vuran: Wireless Sensor Networks.

Das Modul besteht aus einer Vorlesung und einer Übungen. Die in der Vorlesung besprochenen Inhalte werden im Übungsteil anhand von praktischen Anwendungen eingeübt.

Vorkenntnisse

Grundlagen der Analysis, Grundlagen von Rechnernetzen

Qualifikationsziele

Das Modul Sensornetze vermittelt das notwendige Wissen und praktische Fähigkeiten im Umgang mit Sensornetzen, die es den Studenten erlauben, selbstständig Sensornetze und Algorithmen für Sensornetze zu entwickeln, einzusetzen, anzupassen und zu bewerten.

3.3 VT 3: Rechnerarchitektur 2 (INF-RA2)

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tur- nus	Präsenzzeit	Selbst- studium	ECTS
Vorlesung	Vorlesung zur Rechnerarchitektur 2	SoSe	45 h (3 SWS)	75 h	4 CP
Übung	Übungen zur Rechnerarchitektur 2	SoSe	15 h (1 SWS)	45 h	2 CP

Im Modul müssen insgesamt 6 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 4 Wochenstunden. Inklusive Selbststudium sind etwa 180 Stunden aufzuwenden.

Art des Moduls Wahlpflichtmodul mit Pflichtveranstaltungen

Verwendbarkeit Dieses Modul wird in folgenden Studiengängen angeboten
- INF-LGY: Lehramt Gymnasium
- INF-M-120: Masterstudiengang Informatik
- MINF-M-120: Masterstudiengang Medieninformatik

Zeitpunkt im Studienverlauf 2. Semester

Dauer Das Modul erstreckt sich über 1 Semester.

Art der Bewertung benotet

Form der Modulprüfung Klausur (90-180 Minute) oder mündlich (15-30 Minute)
Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine

Modulverantwortlicher Dr. Gordon Cichon

Anbieter Ludwig-Maximilians-Universität München
Fakultät für Mathematik, Informatik und Statistik
Institut für Informatik
Kerninformatik
LFE Mobile und Verteilte Systeme

Unterrichtssprache(n) Deutsch

Inhalte

Das Modul vermittelt ein tiefgehendes Verständnis zu verschiedenen Aspekten leistungsfähiger Computerarchitekturen, wie deren Parallelität (Superskalar, SIMD, Multi- und Many-Core), Speicherhierarchie, die konkrete Umsetzung in verschiedenen Systemen (Compute-Farm, GPU, Cell) und deren Sprachen und APIs (MPI, CUDA, Map-Reduce, etc.). Es werden Anwendungen aus verschiedenen Bereichen besprochen und eingeübt, wie z.B. Wissenschaftliches Rechnen (Simulation in Physik, Elektronik, Mechanik, Strömungsmechanik), Multi-Media (Grafik, 3D-Rendering, Modelle), Eingebettete Systeme (Echtzeit, Mobilfunk, GPS, Steuerung und Regelung, z.B. im Auto), Wirtschaft (Analyse von Finanzprodukten, Millisekundenhandel), Sicherheit (Kryptographie, Bildverarbeitung).

Im Einzelnen werden behandelt:

- Einleitung, Motivation, Geschichte,
- Mathematische Grundlagen,
- Die superskalare Pipeline,
- SIMD-Erweiterungen (MMX, SSE),
- Multi- und Many-Core Systeme I (eng gekoppelte Cluster, GPU),
- Speicherhierarchie I (explizit: GPU, Cell),
- Speicherhierarchie II (implizit: Prediction und Prefetch),
- Multi- und Many-Core Systeme II (lose gekoppelte Cluster, Cloud-Computing).

Literaturhinweise

- Randy Allen and Ken Kennedy, *Optimizing Compilers for Modern Architectures: A Dependence-based Approach*, Morgan Kaufmann, 2001, ISBN-13: 978-1558602861
- David A. Patterson and John L. Hennessy, *Computer Architecture - A Quantitative Approach*, Morgan Kaufmann, 5th Edition, 2011, ISBN-13: 978-0123838728
- Stoer, Bulirsch: *Numerische Mathematik*, Springer Berlin, 10th edition, 2007, ISBN-13: 978-3540453895.

Die Studierenden arbeiten weitgehend selbständig und haben die Möglichkeit sich zu kleinen Teams zusammenschließen. Nach einer Einführung in das jeweilige Themengebiet werden praktische Aufgaben eigenständig gelöst.

Vorkenntnisse

Kenntnisse aus dem Modul Rechnerarchitektur (INF-RA)

Qualifikationsziele

Die Vorlesung vermittelt vertieftes Verständnis von komplexen und leistungsfähigen modernen Rechner, so wie sie heute in fast jedem Computer, Spielkonsole oder Smartphone zu finden sind. Die Studenten erwerben das nötige Wissen, um in der Softwareentwicklung von diesen komplexen Architekturen Gebrauch zu machen und deren volle Leistungsfähigkeit auszuschöpfen.

Die Studenten erhalten dabei Einblicke in Bereiche der Elektrotechnik und Physik.

3.4 VT 4: Praktikum Innovative Mobile Business Applications (INF-MBA)

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tu- nus	Präsenzzeit	Selbst- studium	ECTS
Praktikum	Praktikum Mobile Business Applications	WiSe, SoSe	90 h (6 SWS)	90 h	6 CP

Im Modul müssen insgesamt 6 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 6 Wochenstunden. Inklusive Selbststudium sind etwa 180 Stunden aufzuwenden.

Art des Moduls Wahlpflichtmodul mit Pflichtveranstaltungen

Verwendbarkeit Dieses Modul wird in folgenden Studiengängen angeboten

- INF-B-120: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 60 ECTS-Punkten
- INF-B-150: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 30 ECTS-Punkten
- INF-B-180-CL: Bachelorstudiengang Informatik plus Computerlinguistik
- INF-B-180-MA: Bachelorstudiengang Informatik plus Mathematik
- INF-B-180-STAT: Bachelorstudiengang Informatik plus Statistik
- INF-LGY: Lehramt Gymnasium
- INF-M-120: Masterstudiengang Informatik
- MINF-B-180: Bachelorstudiengang Medieninformatik
- MINF-M-120: Masterstudiengang Medieninformatik

Teilnahmevoraussetzungen keine

Zeitpunkt im Studienverlauf 3. Semester (MINF-M-120, INF-M-120), 4. Semester (INF-B-180-STAT, INF-B-120, INF-B-180-MA, INF-B-150, INF-B-180-CL), 6. Semester (MINF-B-180)

Dauer Das Modul erstreckt sich über 1 Semester.

Art der Bewertung benotet

Form der Modulprüfung Praxisleistung () und mündlich (15-30 Minute)
Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine

Modulverantwortlicher Prof. Dr. Claudia Linnhoff-Popien

Anbieter	Ludwig-Maximilians-Universität München Fakultät für Mathematik, Informatik und Statistik Institut für Informatik Kerninformatik LFE Mobile und Verteilte Systeme
-----------------	--

Unterrichts- sprache(n)	Deutsch
------------------------------------	---------

Inhalte

Das Praktikum findet in Zusammenarbeit mit einem Partner aus der Industrie statt und gliedert sich in zwei Phasen. Die erste Phase bilden insgesamt drei Lehrveranstaltungen während der Vorlesungszeit, in denen die aktuellen Problemstellungen und Themen vorgestellt werden sowie Teams zur Bearbeitung dieser Themen gebildet werden. Die Teilnehmer haben hierbei die Möglichkeit, über die Themenvorschläge zu diskutieren und ihre eigenen Ideen und Vorschläge einzubringen. Schließlich werden Aufgaben verteilt und die entsprechenden Konzepte dazu entwickelt.

Die zweite Phase stellt schließlich die praktische Umsetzung der erarbeiteten Konzepte dar. Über den Zeitraum von zwei Wochen wird (wenn möglich in den Räumlichkeiten des Industriepartners) in Gruppen entwickelt und implementiert. Das Praktikum wird mit einer Präsentation abgeschlossen, bei der die Teams die Möglichkeit haben, ihre Ergebnisse vorzustellen.

Typischerweise beinhaltet das Praktikum daher die folgenden Aspekte:

- mobile Anwendungsentwicklung (u.a. für iOS oder Android Geräte),
- Implementierung passender Datenbank- und Backend-Systeme (meist Java-basiert),
- Umsetzung Hardware-naher Funktionalitäten auf speziellen Systemen, z.B. RaspberryPi oder Arduino-Boards.

Die Teilnehmer arbeiten selbständig in Teams von meist vier bis sechs Personen und werden dabei intensiv von Mitarbeitern des Lehrstuhls und des Industriepartners betreut.

Vorkenntnisse

Fundierte Kenntnisse der objektorientierten Programmierung und verteilter Systeme.

Qualifikationsziele

Das Praktikum bietet den Teilnehmern die Möglichkeit, innovative Lösungskonzepte für aktuelle Problemstellungen in Zusammenarbeit mit einem Partner aus der Industrie zu konzipieren und diese prototypisch umzusetzen. Praktische Erfahrung bei der Durchführung innovativer IT-Projekte soll vermittelt werden. Dabei sind die Studenten auf allen Ebenen gefordert: (Nicht vorhersehbare) technische Herausforderungen und Probleme müssen gelöst werden, die Teams müssen sich intern organisieren und sie müssen vor allem auch unter Stress als solches funktionieren. Die Teilnehmer lernen, ihre vorhandenen Fähigkeiten im Bereich der Software-Entwicklung gezielt einzusetzen, und sich noch nicht vorhandene Fähigkeiten eigenständig und schnell anzueignen.

Neben der Herausforderung, unbekannte Probleme zu lösen und sich selbständig in neue Technologien einzuarbeiten, sind auch technikfernere Aufgaben wie das Projektmanagement, die Kommunikation mit realen Kunden und schließlich auch die Präsentation der Ergebnisse zu meistern.

3.5 VT 5: Praktikum für iOS Entwicklung (INF-IOS)

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tu- rus	Präsenzzeit	Selbst- studium	ECTS
Vorlesung	Vorlesung iOS Entwicklung	WiSe, SoSe	15 h (1 SWS)	45 h	2 CP
Praktikum	Praktikum für iOS Entwicklung	WiSe, SoSe	45 h (3 SWS)	75 h	4 CP

Im Modul müssen insgesamt 6 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 4 Wochenstunden. Inklusive Selbststudium sind etwa 180 Stunden aufzuwenden.

Art des Moduls Wahlpflichtmodul mit Pflichtveranstaltungen

Verwendbarkeit Dieses Modul wird in folgenden Studiengängen angeboten

- INF-B-120: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 60 ECTS-Punkten
- INF-B-150: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 30 ECTS-Punkten
- INF-B-180-CL: Bachelorstudiengang Informatik plus Computerlinguistik
- INF-B-180-MA: Bachelorstudiengang Informatik plus Mathematik
- INF-B-180-STAT: Bachelorstudiengang Informatik plus Statistik
- INF-LGY: Lehramt Gymnasium
- INF-M-120: Masterstudiengang Informatik
- MINF-M-120: Masterstudiengang Medieninformatik
- MINF-M-120-KW: Masterstudiengang Medieninformatik mit Anwendungsfach Kommunikationswissenschaft

Teilnahmevoraussetzungen keine

Zeitpunkt im Studienverlauf 3. Semester (MINF-M-120-KW, MINF-M-120, INF-M-120), 5. Semester (INF-B-180-STAT, INF-B-120, INF-B-180-MA, INF-B-150, INF-B-180-CL)

Dauer Das Modul erstreckt sich über 1 Semester.

Art der Bewertung benotet

Form der Modulprüfung Praxisleistung ()
Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine

**Modul-
verantwort-
licher** Prof. Dr. Claudia Linnhoff-Popien

Anbieter Ludwig-Maximilians-Universität München
Fakultät für Mathematik, Informatik und Statistik
Institut für Informatik
Kerninformatik

**Unterrichts-
sprache(n)** Deutsch

Inhalte

Das Praktikum besteht aus einer Theorie- und einer Programmierphase. Beginnend mit der Theoriephase werden Grundlagen der Entwicklung für das iOS-Betriebssystem vermittelt. Dazu gehört eine Einführung in die Programmiersprache Objective-C und objektorientierte Programmierung mit Message Passing ebenso dazu wie ein Überblick über die Architektur des Betriebssystems. Dabei werden wichtige Aspekte des Foundation Frameworks und einiger ausgewählter Core-Frameworks vorgestellt. Die Theorieveranstaltungen liefern vor allem die zentralen Ideen und Begriffe, die den Teilnehmern einen tieferen Einstieg in die jeweiligen Themenbereiche erleichtern. In der Praxisphase werden eigenständige iOS-Applikationen in Teams von 3 bis 6 Teilnehmern konzipiert und entwickelt. Dabei sollen die Teilnehmer auf die vermittelten Grundlagen aufbauen können. Der Theorieteil der Veranstaltung besteht aus einer interaktiven Vorlesung. In der Praxisphase arbeiten die Teilnehmer dann selbstständig in kleinen Teams.

Inhalte der Vorlesung

- Einführung in Objective-C,
- Modell-View-Controller Konzept,
- User-Interface-Komponenten,
- User-Interface-Navigation,
- Storyboards,
- Benachrichtigungen,
- iOS-Zustandsmodell,
- Speichern von Zuständen / Daten,
- Behandlung von Events (Touches und Gesten),
- Verwendung von Sensoren,
- Verwendung von Built-In Datenbanken,
- Verwendung von Kommunikationsschnittstellen und Bibliotheken,
- Threading und Dispatch-Queues.

Literaturhinweise

- iOS Programming - The Big Nerd Ranch Guide (4th Edition) (Big Nerd Ranch Guides)

Der Theorieteil der Veranstaltung besteht aus einer interaktiven Vorlesung. In der Praxisphase arbeiten die Teilnehmer dann selbstständig in kleinen Teams.

Vorkenntnisse

Fundierte Kenntnisse der objektorientierten Programmierung.

Qualifikationsziele

Das Praktikum bietet einen Einstieg in die iOS-Entwicklung mit Objective-C. Die Teilnehmer sollen die Fähigkeit entwickeln, sich in einer weitgehend unbekanntem Programmiersprache und einem ebensolchen Betriebssystem soweit zügig einzuarbeiten, dass sie ihre Ideen uneingeschränkt umsetzen können.

3.6 VT 6: Automotive Praktikum (INF-AP)

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tur- nus	Präsenzzeit	Selbst- studium	ECTS
Praktikum	Automotive Praktikum	WiSe	30 h (2 SWS)	60 h	3 CP

Im Modul müssen insgesamt 3 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 2 Wochenstunden. Inklusive Selbststudium sind etwa 90 Stunden aufzuwenden.

**Art des Mo-
duls** Wahlpflichtmodul mit Pflichtveranstaltungen

**Verwendbar-
keit** Dieses Modul wird in folgenden Studiengängen angeboten
- INF-LGY: Lehramt Gymnasium
- INF-M-120: Masterstudiengang Informatik
- MINF-M-120: Masterstudiengang Medieninformatik

**Zeitpunkt
im Studien-
verlauf** 3. Semester

Dauer Das Modul erstreckt sich über 1 Semester.

**Art der Be-
wertung** benotet

Form der Hausarbeit () und Praxisleistung ()
Modulprüfung Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine

**Modul-
verantwort-
licher** Dr. Gordon Cichon

Anbieter Ludwig-Maximilians-Universität München
Fakultät für Mathematik, Informatik und Statistik
Institut für Informatik
Kerninformatik
LFE Mobile und Verteilte Systeme

**Unterrichts-
sprache(n)** Deutsch

Inhalte

Das Automotive Blockpraktikum vermittelt eine praktische Einführung in die Anwendung von eingebetteten Mikroprozessorsystemen im Automotive-Umfeld insbesondere in den Bereichen Powertrain und Safety. Hierbei wird vertieft auf die Anforderungen im Echtzeitbetrieb und die Einschränkungen durch Stromverbrauch und Arbeitsspeicher eingegangen. Den Studenten wird die Möglichkeit geboten mit realen Mikroprozessoren aus dem Automotive-Umfeld zu arbeiten. Nach deren Inbetriebnahme wird darauf eingegangen, wie diese zur Digital und Analog Wandlung sowie zur Ein- und Ausgabe von Daten eingesetzt werden können. Ein erstes System bestehend aus Sensoren und Aktoren wird von den Studenten aufgesetzt. Zuletzt wird praktisch vermittelt, wie eine geräteübergreifende Kommunikation mittels des CAN-Busses realisiert werden kann.

Inbesondere werden folgende Inhalte praxisnah behandelt:

- Einführung und Inbetriebnahme eines Systems,
- Mikroarchitektur von Echtzeitsystemen und spezifische Optimierungen für eingebettete Anwendungen,
- Peripherie: Ein- und Ausgabe, Digital und Analog Wandlung,
- Sensoren und Aktoren,
- Kommunikation mit CAN-Bus.

Die Studierenden arbeiten weitgehend selbständig und haben die Möglichkeit sich zu kleinen Teams zusammenschließen. Nach einer Einführung in das jeweilige Themengebiet werden praktische Aufgaben eigenständig gelöst.

Vorkenntnisse

Grundlegende Kenntnisse in der C-Programmierung; Vorlesung Rechnerarchitektur

Qualifikationsziele

Das Automotive Blockpraktikum vermittelt das Wissen über eingebettete Mikrocontroller-Systeme im Automotive Umfeld. Den Teilnehmern wird die Fertigkeit vermittelt, mit den speziellen Anforderungen des Echtzeitbetriebs und den Einschränkungen in Strom- und Speicherverbrauch umzugehen. Die Veranstaltung fördert die soziale Kompetenz, in Projekten der Automobilindustrie zur Softwareentwicklung beizutragen.

Die Studenten erhalten darüber hinaus Einblicke in Bereiche der Elektrotechnik und Physik.

3.7 VT 7: Geistiges Eigentum für die Informationstechnologie (INF-IPIT)

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tu- nus	Präsenzzeit	Selbst- studium	ECTS
Vorlesung	Vorlesung zu Geistiges Eigentum für die Informationstechnologie	SoSe	22.5 h (1.5 SWS)	37.5 h	2 CP
Übung	Übungen zu Geistiges Eigentum für die Informationstechnologie	SoSe	7.5 h (0.5 SWS)	22.5 h	1 CP

Im Modul müssen insgesamt 3 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 2 Wochenstunden. Inklusiv Selbststudium sind etwa 90 Stunden aufzuwenden.

Art des Moduls Wahlpflichtmodul mit Pflichtveranstaltungen

Verwendbarkeit Dieses Modul wird in folgenden Studiengängen angeboten

- INF-B-120: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 60 ECTS-Punkten
- INF-B-150: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 30 ECTS-Punkten
- INF-B-180-CL: Bachelorstudiengang Informatik plus Computerlinguistik
- INF-B-180-MA: Bachelorstudiengang Informatik plus Mathematik
- INF-B-180-STAT: Bachelorstudiengang Informatik plus Statistik
- INF-LGY: Lehramt Gymnasium
- INF-M-120: Masterstudiengang Informatik
- MINF-M-120: Masterstudiengang Medieninformatik

Teilnahmevoraussetzungen keine

Zeitpunkt im Studienverlauf 2. Semester (MINF-M-120, INF-M-120), 4. Semester (INF-B-180-STAT, INF-B-120, INF-B-180-MA, INF-B-150, INF-B-180-CL)

Dauer Das Modul erstreckt sich über 1 Semester.

Art der Bewertung benotet

Form der Modulprüfung Klausur (90-180 Minute)
Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine

Modulverantwortlicher Prof. Dr. Claudia Linnhoff-Popien

Anbieter	Ludwig-Maximilians-Universität München Fakultät für Mathematik, Informatik und Statistik Institut für Informatik Kerninformatik LFE Mobile und Verteilte Systeme
-----------------	--

Unterrichts- sprache(n)	Deutsch
------------------------------------	---------

Inhalte

Das Modul vermittelt einen Überblick über die möglichen Schutzrechte des gewerblichen Rechtsschutzes und dient auch dazu, konkrete Handlungsoptionen für den Schutz von Entwicklungen auf dem Gebiet der Informatik aufzuzeigen. Neben nicht-technischen Schutzrechten, wie dem Markenrecht, dem Geschmacksmusterrecht, dem Urheberrecht, werden die technischen Schutzrechte, das Patentrecht und das Gebrauchsmusterrecht schwerpunktmäßig behandelt werden. Neben der Frage „Wie kann ich meine Entwicklungsleistung sinnvoll vor Nachahmung schützen?“ wird auch behandelt, wie sich ein erworbenes Schutzrecht durchsetzen lässt.

Im Einzelnen werden behandelt:

- Technische Schutzrechte (Patente, Gebrauchsmuster, Schutzrecht-Strategien),
- Computer-implementierte Erfindungen (Urberschutz und Patentschutz, Lizenzverträge, Patentschutz für computer-implementierte Erfindungen),
- Marken (Kennzeichenrechte, Marken, Markenschutz, Schutzvoraussetzungen, Markenverletzungen),
- Geschmacksmuster, Arbeitnehmererfinderrecht,
- Verletzung von Schutzrechten.

Literaturhinweise:

- Andreas Heinemann, Patent- und Designrecht: PatR, 12. Auflage, ISBN-13: 978-3-406-66154-9,
- Volker Ilzhöfer und Rainer Engels, Patent-, Marken- und Urheberrecht: Leitfaden für Ausbildung und Praxis, 8. Auflage, ISBN-13: 978-3800637270,
- Fachzeitschriften: "Mitteilungen der deutschen Patentanwälte", "GRUR", "GRUR Int.", "Computer und Recht".

Das Modul besteht aus einer Vorlesung und einer Übung. Die in der Vorlesung besprochenen Inhalte werden im Übungsteil anhand von praktischen Fällen vertieft. Dabei werden teilweise auch neue Inhalte erörtert oder – je nach Teilnehmeranzahl – in praktischen Übungen herausgearbeitet.

Qualifikationsziele

Die Studierenden erwerben ein Verständnis der Grundzüge und der Möglichkeiten im gewerblichen Rechtsschutz. Insbesondere wird Hintergrundwissen zu sinnvollen Schutzmöglichkeiten für die Informatik mit Bezug auf das spannungsreiche Thema der *Software-Patente* aufgezeigt. Der Student erwirbt die nötigen Fähigkeiten, um solche Schutzmöglichkeiten in der Informatik verstehen und bewerten zu können.

3.8 VT 8: Mobile und Verteilte Systeme (INF-MVS)

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tu- rus	Präsenzzeit	Selbst- studium	ECTS
Vorlesung	Vorlesung zu Mobile und Verteilte Systeme	WiSe, SoSe	30 h (2 SWS)	30 h	2 CP
Praktikum	Praxisteil zu Mobile und Verteilte Systeme	WiSe, SoSe	60 h (4 SWS)	60 h	4 CP

Im Modul müssen insgesamt 6 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 5 Wochenstunden. Inklusiv Selbststudium sind etwa 180 Stunden aufzuwenden.

Art des Moduls	Wahlpflichtmodul mit Pflichtveranstaltungen
-----------------------	---

Verwendbarkeit	Dieses Modul wird in folgenden Studiengängen angeboten - INF-LGY: Lehramt Gymnasium - INF-M-120: Masterstudiengang Informatik - MINF-M-120: Masterstudiengang Medieninformatik - MINF-M-120-KW: Masterstudiengang Medieninformatik mit Anwendungsfach Kommunikationswissenschaft - MINF-M-120-MCI: Masterstudiengang Mensch-Computer Interaktion - MINF-M-120-MG: Masterstudiengang Medieninformatik mit Anwendungsfach Mediengestaltung - MINF-M-120-MW: Masterstudiengang Medieninformatik mit Anwendungsfach Medienwirtschaft
-----------------------	---

Teilnahmevoraussetzungen	keine
---------------------------------	-------

Zeitpunkt im Studienverlauf	2. Semester
------------------------------------	-------------

Dauer	Das Modul erstreckt sich über 1 Semester.
--------------	---

Art der Bewertung	benotet
--------------------------	---------

Form der Modulprüfung	Klausur (90-180 Minute) oder mündlich (15-30 Minute) Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine
------------------------------	--

Modulverantwortlicher	Prof. Dr. Claudia Linnhoff-Popien
------------------------------	-----------------------------------

Anbieter Ludwig-Maximilians-Universität München
Fakultät für Mathematik, Informatik und Statistik
Institut für Informatik
Kerninformatik

**Unterrichts-
sprache(n)** Deutsch

Inhalte

Das Modul stellt eine Einführung in den Bereich der Verteilten Systeme unter besonderer Berücksichtigung von Ubiquitous Computing Umgebungen dar. Behandelt werden im Einzelnen:

- Charakterisierung Mobiler und Verteiler Systeme,
- Mobile Endgeräte und Betriebssysteme,
- Kommunikation in Verteilten Systemen,
- Naming-, Directory und Lokalisierungsdienste,
- Dienste und Dienstvermittlung,
- Kontextsensitive Dienstnutzung,
- Synchronisation und Abstimmung in Verteilten Systemen,
- Sicherheit in Verteilten Systemen,
- Skalierbarkeit mittels Replikation, Caching und Verteilung.

Literaturhinweise:

- George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair, Distributed Systems: Concepts and Design, Addison-Wesley, 5th Ed., 2011, ISBN-13 978-0132143011,
- Andrew S. Tanenbaum, Distributed Systems: Principles and Paradigms, Prentice Hall, 2nd rev. ed., 2006, ISBN-13: 978-0132392273,
- Alexander Schill, Thomas Springer, Verteilte Systeme: Grundlagen und Basistechnologien, Springer, 2nd. ed., 2012, ISBN-13: 978-3642257957.

Das Modul besteht aus einer Vorlesung sowie einer zentralen Hörsaalübung. Die in der Vorlesung besprochenen Inhalte werden im Übungsteil anhand von praktischen Anwendungen eingeübt.

Vorkenntnisse

Vorkenntnisse in der Software-Entwicklung mit Java sind sehr hilfreich.

Qualifikationsziele

Dieses Modul vermittelt den Studierenden eine umfassende Einführung in mobile und verteilte Systeme. Dies ist ein wichtiger Schlüssel zur zielgerichteten Anwendung und Entwicklung solcher Systeme. Es werden die nötigen Grundkenntnisse geschaffen, so dass sich die Studenten selbst in weiterführende Konzepte selbstständig einarbeiten können.

Die Studenten lernen sich schnell und umfangreich in komplexe Systeme und Zusammenhänge einzuarbeiten.

3.9 VT 9: Methoden des Software Engineering (INF-MSE)

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tu- rus	Präsenzzeit	Selbst- studium	ECTS
Vorlesung	Vorlesung Methoden des Software Engineering	WiSe	45 h (3 SWS)	75 h	4 CP
Übung	Übungen zu Methoden des Software Engineering	WiSe	30 h (2 SWS)	30 h	2 CP

Im Modul müssen insgesamt 6 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 5 Wochenstunden. Inklusive Selbststudium sind etwa 180 Stunden aufzuwenden.

Art des Moduls	Wahlpflichtmodul mit Pflichtveranstaltungen
-----------------------	---

Verwendbarkeit	<p>Dieses Modul wird in folgenden Studiengängen angeboten</p> <ul style="list-style-type: none"> - INF-B-120: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 60 ECTS-Punkten - INF-B-150: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 30 ECTS-Punkten - INF-B-180-CL: Bachelorstudiengang Informatik plus Computerlinguistik - INF-B-180-MA: Bachelorstudiengang Informatik plus Mathematik - INF-B-180-STAT: Bachelorstudiengang Informatik plus Statistik - INF-LGY: Lehramt Gymnasium - INF-M-120: Masterstudiengang Informatik - MINF-B-180: Bachelorstudiengang Medieninformatik - MINF-M-120: Masterstudiengang Medieninformatik - MINF-M-120-KW: Masterstudiengang Medieninformatik mit Anwendungsfach Kommunikationswissenschaft - MINF-M-120-MCI: Masterstudiengang Mensch-Computer Interaktion - MINF-M-120-MG: Masterstudiengang Medieninformatik mit Anwendungsfach Mediengestaltung - MINF-M-120-MW: Masterstudiengang Medieninformatik mit Anwendungsfach Medienwirtschaft
-----------------------	--

Teilnahmevoraussetzungen	keine
---------------------------------	-------

Zeitpunkt im Studienverlauf	1. Semester (MINF-M-120-KW, MINF-M-120, INF-M-120, MINF-M-120-MCI, MINF-M-120-MG, MINF-M-120-MW), 3. Semester (INF-M-120), 5. Semester (INF-B-180-STAT, INF-B-120, INF-B-180-MA, INF-B-150, INF-B-180-CL, MINF-B-180)
------------------------------------	---

Dauer	Das Modul erstreckt sich über 1 Semester.
--------------	---

Art der Bewertung	benotet
--------------------------	---------

Form der Klausur (90-180 Minute) oder mündlich (15-30 Minute)
Modulprüfung Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine

Modulverantwortlicher Studiengangskoordinator(INF-LGY)

Anbieter Ludwig-Maximilians-Universität München
Fakultät für Mathematik, Informatik und Statistik
Institut für Informatik
Kerninformatik

Unterrichtssprache(n) Deutsch

Inhalte

Software-Engineering ist die Disziplin der ingenieurmäßigen Herstellung großer Softwaresysteme. Dies beinhaltet die Bereitstellung und systematische Verwendung von Methoden, Verfahren und Werkzeugen zur Entwicklung, zum Betrieb und zur Wartung von Software. Das Modul beschäftigt sich mit dem vollständigen Prozess der Software-Entwicklung von den Anforderungen über die Software-Architektur bis zu Verifikation, Validierung und Test. Insbesondere werden die Themen formale Methoden und Software-Entwicklungsprozesse behandelt. Als grafische Modellierungssprache wird die Unified Modeling Language (UML) verwendet. Das Modul verbindet praktische Inhalte mit den theoretischen Grundlagen der Software-Entwicklung.

Das Modul besteht aus einer Vorlesung sowie Übungen in Gruppen. Die in der Vorlesung besprochenen Inhalte werden in den Übungen anhand von praktischen Anwendungen eingeübt. Dabei werden spezielle Software-Entwicklungsaufgaben mit systematischen Methoden gelöst.

Qualifikationsziele

Die Studierenden sollen einen Überblick über die wichtigsten Vorgehensweisen, Methoden und Techniken zur systematischen Entwicklung von Softwaresystemen erhalten. Sie sollen in die Lage versetzt werden, Lösungsansätze für praktische Software-Entwicklungsprobleme vorschlagen zu können und auf systematische Art und Weise umsetzen zu können.

3.10 VT 10: Formale Techniken in der Software-Entwicklung (INF-FTS)

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tu- rus	Präsenzzeit	Selbst- studium	ECTS
Vorlesung	Vorlesung Formale Techniken in der Software-Entwicklung	SoSe	45 h (3 SWS)	45 h	3 CP
Übung	Übungen zu Formale Techniken in der Software-Entwicklung	SoSe	30 h (2 SWS)	60 h	3 CP

Im Modul müssen insgesamt 6 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 5 Wochenstunden. Inklusiv Selbststudium sind etwa 180 Stunden aufzuwenden.

Art des Moduls Wahlpflichtmodul mit Pflichtveranstaltungen

Verwendbarkeit Dieses Modul wird in folgenden Studiengängen angeboten

- INF-LGY: Lehramt Gymnasium
- INF-M-120: Masterstudiengang Informatik
- MINF-M-120: Masterstudiengang Medieninformatik
- MINF-M-120-KW: Masterstudiengang Medieninformatik mit Anwendungsfach Kommunikationswissenschaft
- MINF-M-120-MCI: Masterstudiengang Mensch-Computer Interaktion
- MINF-M-120-MG: Masterstudiengang Medieninformatik mit Anwendungsfach Mediengestaltung
- MINF-M-120-MW: Masterstudiengang Medieninformatik mit Anwendungsfach Medienwirtschaft

Teilnahmevoraussetzungen keine

Zeitpunkt im Studienverlauf 1. Semester (MINF-M-120-KW, MINF-M-120, MINF-M-120-MCI, MINF-M-120-MG, MINF-M-120-MW), 2. Semester (INF-M-120)

Dauer Das Modul erstreckt sich über 1 Semester.

Art der Bewertung benotet

Form der Modulprüfung Klausur (90-180 Minute) oder mündlich (15-30 Minute)
Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine

Modulverantwortlicher Studiengangskoordinator(INF-LGY)

Anbieter Ludwig-Maximilians-Universität München
Fakultät für Mathematik, Informatik und Statistik
Institut für Informatik
Kerninformatik

**Unterrichts-
sprache(n)** Deutsch

Inhalte

Formale Techniken zur Systementwicklung basieren auf mathematisch fundierten Beschreibungstechniken und Vorgehensweisen. Die mathematische Fundierung erlaubt es, statische und dynamische Eigenschaften von Modellen und Programmen präzise zu definieren und liefert somit die Voraussetzung für viele Validierungs-, Verifikations- und Verfeinerungstechniken. Das Modul gibt eine Einführung in eine oder mehrere der folgenden formalen Methoden der Software-Entwicklung: formale objekt-orientierte Software-Entwicklung, daten-orientierte Spezifikationsentwicklung, Modellierung und Validierung nebenläufiger reaktiver Systeme, Modelchecking, sowie Analyse nicht-funktionaler Eigenschaften wie Performanz.

Das Modul besteht aus einer Vorlesung sowie Übungen in kleinen Gruppen. Die in der Vorlesung besprochenen Inhalte werden im Übungsteil anhand von praktischen Anwendungen eingeübt.

Qualifikationsziele

Die Studierenden sollen ausgewählte formale Spezifikations- und Verifikationstechniken kennenlernen und verstehen, deren Möglichkeiten ausnützen können und in konkreten Fallbeispielen anwenden können.

3.11 VT 11: Software Engineering für spezielle Anwendungsgebiete (INF-SEspA)

Zugeordnete Modulteile:

Lehrform	Veranstaltung	Tu- rus	Präsenzzeit	Selbst- studium	ECTS
Vorlesung	Vorlesung Software Engineering für spezielle Anwendungsgebiete	SoSe	45 h (3 SWS)	75 h	4 CP
Übung	Übungen zu Software Engineering für spezielle Anwendungsgebiete	SoSe	30 h (2 SWS)	30 h	2 CP

Im Modul müssen insgesamt 6 ECTS-Punkte erworben werden. Die Präsenzzeit beträgt 5 Wochenstunden. Inklusiv Selbststudium sind etwa 180 Stunden aufzuwenden.

Art des Moduls	Wahlpflichtmodul mit Pflichtveranstaltungen
-----------------------	---

Verwendbarkeit	<p>Dieses Modul wird in folgenden Studiengängen angeboten</p> <ul style="list-style-type: none"> - INF-B-120: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 60 ECTS-Punkten - INF-B-150: Bachelorstudiengang Informatik mit Nebenfach im Umfang von 30 ECTS-Punkten - INF-B-180-CL: Bachelorstudiengang Informatik plus Computerlinguistik - INF-B-180-MA: Bachelorstudiengang Informatik plus Mathematik - INF-B-180-STAT: Bachelorstudiengang Informatik plus Statistik - INF-LGY: Lehramt Gymnasium - INF-M-120: Masterstudiengang Informatik - MINF-M-120: Masterstudiengang Medieninformatik - MINF-M-120-KW: Masterstudiengang Medieninformatik mit Anwendungsfach Kommunikationswissenschaft - MINF-M-120-MCI: Masterstudiengang Mensch-Computer Interaktion - MINF-M-120-MG: Masterstudiengang Medieninformatik mit Anwendungsfach Mediengestaltung - MINF-M-120-MW: Masterstudiengang Medieninformatik mit Anwendungsfach Medienwirtschaft
-----------------------	--

Teilnahmevoraussetzungen	keine
---------------------------------	-------

Zeitpunkt im Studienverlauf	2. Semester (MINF-M-120-KW, MINF-M-120, INF-M-120, MINF-M-120-MCI, MINF-M-120-MG, MINF-M-120-MW), 4. Semester (INF-B-180-STAT, INF-B-120, INF-B-180-MA, INF-B-150, INF-B-180-CL)
------------------------------------	--

Dauer	Das Modul erstreckt sich über 1 Semester.
--------------	---

Art der Bewertung	benotet
--------------------------	---------

Form der Klausur (90-180 Minute) oder mündlich (15-30 Minute)
Modulprüfung Wiederholbarkeit: beliebig, Zulassungsvoraussetzung: keine

Modulverantwortlicher Prof. Dr. Rolf Hennicker

Anbieter Ludwig-Maximilians-Universität München
Fakultät für Mathematik, Informatik und Statistik
Institut für Informatik
Kerninformatik
LFE Programmierung und Softwaretechnik

Unterrichtssprache(n) Deutsch

Inhalte

Für die unterschiedlichen Programmierparadigmen und die vielfältigen Einsatzgebiete von Softwaresystemen benötigt man speziell abgestimmte Entwicklungstechniken. Dieses Modul gibt einen Einblick in Software Engineering Methoden für spezielle Anwendungsgebiete. Insbesondere werden nebenläufige und verteilte Systeme, eingebettete Systeme, Webanwendungen, sowie Systeme, die von nicht-funktionalen Eigenschaften wie Performanz und Sicherheit abhängen, betrachtet.

Der Modul besteht aus einer Vorlesung und Übungen in Gruppen. Die in der Vorlesung besprochenen Inhalte werden in den Übungen anhand von speziellen Software-Entwicklungsaufgaben eingeübt.

Qualifikationsziele

Die Studierenden sollen systematische Software-Entwicklungstechniken für eines der oben genannten Anwendungsgebiete kennen lernen und diese an praktischen Beispielen anwenden können. Sie sollen einen Überblick über die grundlegenden Software Engineering Methoden dieses Anwendungsgebiets erhalten und in die Lage versetzt werden, für praktische Fragestellungen Lösungsansätze vorschlagen und beurteilen zu können.