

<b>Amtliche Mitteilungen der</b>	
<b>Philipps</b>	
	<b>Universität Marburg</b>
<b>Veröffentlichungsnummer: 70/2010</b>	<b>Veröffentlicht am: 18.11.2010</b>

Das Zentrum für Lehrerbildung der Philipps-Universität Marburg hat gemäß § 48 Abs. 2 Nr. 1 Hessisches Hochschulgesetzes (HHG) in der Fassung vom 14. Dezember 2009 (GVBl. I S. 666) im Benehmen mit dem Fachbereichsrat Mathematik und Informatik der Philipps-Universität Marburg folgende fachspezifische Bestimmungen für das Fach „Informatik“ im Studiengang „Lehramt an Gymnasien“ an der Philipps-Universität Marburg beschlossen. Diese sind als Ziffer 11 Bestandteil des Anhangs 3 der Allgemeinen Bestimmungen für das modulare Studium „Lehramt an Gymnasien“ vom 03. März 2010:

## **11. Kerncurriculum Informatik**

Das Studium des Kerncurriculums Informatik soll die Studierenden auf ihre Tätigkeit als Informatiklehrer sowohl fachlich als auch fachdidaktisch vorbereiten. Lehrer des Fachs Informatik sollen befähigt sein, den Schülern wesentliche Denkweisen der Informatik zu vermitteln und sie in den Kenntnisstand zu versetzen, Methoden und Verfahren der Informatik in angemessener Weise konkret anzuwenden. Im Mittelpunkt des Lehramtsstudiums stehen die Grundlagen und konsolidierten Teilbereiche der Informatik, aber auch neue Entwicklungen können im Rahmen eines Vertiefungsmoduls berücksichtigt werden. Außerdem soll der angehende Lehrer lernen, informatikspezifische Fachinhalte didaktisch aufzubereiten und im Unterricht zu vermitteln. Dazu gehören auch Kenntnisse zur Beurteilung von Rechnerausstattungen und die Planung von Rechneinsatz an Schulen unter pädagogischen Gesichtspunkten.

### **11.1 Modulliste**

#### **Pflichtmodule**

##### **Fachwissenschaftliche Module:**

- CS 110 – Praktische Informatik I: Programmierung (9 LP)
- CS 180 – Mathematik I (9 LP)
- CS 380 – Logik und Diskrete Mathematik für Informatiker (9 LP)

##### **Schnittstellenmodule (integrierte fachwissenschaftlich-didaktische Module):**

- CS 210 – Praktische Informatik II: Algorithmen und Datenstrukturen (9 LP)
- CS 310 – Praktische Informatik III: Konzepte von Programmiersprachen (8 LP)
- CS 460 – Theoretische Informatik (9 LP)
- CS 410 – Datenbanksysteme (8 LP)

##### **Fachdidaktikmodule**

- CS 505 – Einführung in die Didaktik der Informatik (3 LP)

#### **Wahlpflichtmodule**

##### **Fachwissenschaftliche Wahlpflichtmodule**

- **Bereich Systementwicklung**

CS 340 – Einführung in die Softwaretechnik	(7 LP)
CS 420 – Software-Praktikum (Schnittstellenmodul (4 LP FW / 4 LP FD))	(8 LP)
CS 501 – Programmieren in C++	(8 LP)
CS 507 – Moderne Methoden der Systementwicklung	(7 LP)
CS 531 – Systemanalyse und Modellierung	(7 LP)
CS 532 – Entwurf großer Softwaresysteme	(7 LP)
CS 533 – Software-Projektmanagement und Qualitätssicherung	(7 LP)
CS 634 – Modellierung von Informationssystemen und Wissensrepräsentation	(5 LP)
CS 551 – Grundlagen des Compilerbaus	(9 LP)
CS 553 – Parallele Programmierung	(9 LP)
CS 561 – Modellgetriebene Software-Entwicklung	(9 LP)
CS 562 – Visuelle Sprachen	(6 LP)
CS 607 – Moderne Methoden der Systementwicklung II	(7 LP)
CS 652 – Parallele und verteilte Algorithmen	(8 LP)
CS 653 – Parallelität in funktionalen Programmiersprachen	(8 LP)
CS 661 – Softwarequalität	(6 LP)
CS 581 – Grafikprogrammierung 1	(9 LP)
CS 681 – Grafikprogrammierung 2	(9 LP)
○ <b>Bereich Systemumgebung</b>	
CS 140 – Technische Informatik I (Rechnerstrukturen)	(9 LP)
CS 240 – Techn. Inf. II (Betriebssysteme, Rechnerkomm.)	(9 LP)
CS 508 – Aktuelle Methoden zur Informationsverarbeitung	(7 LP)
CS 511 – Rechnernetze	(9 LP)
CS 512 – Betriebssysteme	(6 LP)
CS 513 – Verteilte Systeme	(6 LP)
CS 514 – Grid Computing	(6 LP)
CS 541 – Einführung in die Bioinformatik	(9 LP)
CS 542 – Maschinelles Lernen	(9 LP)
CS 543 – Computational Intelligence	(9 LP)
CS 544 – Fuzzy Systeme	(6 LP)
CS 571 – Index- und Speicherstrukturen	(6 LP)
CS 572 – Information Retrieval	(6 LP)
CS 573 – Geo-Datenbanken	(6 LP)
CS 591 – Knowledge Discovery	(7 LP)
CS 592 – Einführung in die Künstliche Intelligenz	(6 LP)
CS 593 – Neuronale Netze	(6 LP)

CS 608 – Aktuelle Methoden zur Informationsverarbeitung II	(7 LP)
CS 671 – Datenintegration	(6 LP)
CS 682 – Multimediakommunikation	(9 LP)
CS 691 – Temporales Data Mining	(7 LP)
CS 692 – Künstliche Intelligenz II	(6 LP)

○ **Bereich Theoretische Informatik**

CS 509 – Formale Methoden und Grundlagen der Programmierung	(7 LP)
CS 521 – Model Checking	(9 LP)
CS 522 – Rechnergestützte Beweissysteme	(9 LP)
CS 523 – Berechenbarkeit und Beweissysteme	(9 LP)
CS 552 – Semantik von Programmiersprachen	(8 LP)
CS 609 – Formale Methoden und Grundlagen der Programmierung II	(7 LP)
CS 621 – Abstrakte Datentypen	(9 LP)
CS 622 – Zustandsbasierte Systeme	(9 LP)
CS 651 – Strukturen funktionaler Programmiersprachen	(8 LP)

In den Bereich Theoretische Informatik können auch die folgenden im Bereich Systementwicklung aufgeführten Module eingebracht werden:

CS 551 – Grundlagen des Compilerbaus	(9 LP)
CS 553 – Parallele Programmierung	(9 LP)
CS 652 – Parallele und verteilte Algorithmen	(8 LP)
CS 653 – Parallelität in funktionalen Programmiersprachen	(8 LP)

○ **Seminar**

CS 600 – Vertiefung / Spezielle Themen der Informatik (Seminar)	(3 LP)
---	--------

**Fachdidaktikmodule**

○ CS 601 – Didaktik der Informatik – Vertiefung	(3 LP)
○ CS 602 – Schulpraktische Studien II im Fach Informatik	(5 LP)
○ CS 603 – Kompensationsmodul zu Schulpraktischen Studien II im Fach Informatik	(5 LP)
○ CS 604 – Proseminar in Informatik	(4 LP)
○ CS 605 – Tutorenseminar in Informatik	(4 LP)
○ CS 502 – Information, Informationssysteme, Informationsgesellschaft	(4 LP)
○ CS 503 – Neuere Entwicklungen der Fachdidaktik Informatik	(4 LP)
○ CS 606 – Gesellschaftliche Aspekte der Informatik	(4 LP)

## 11.2 Studien- und Modulverlaufsplan

### Aufgliederung der Studienleistung

Gesamtumfang 90 LP = 60 LP Fachwissenschaft (FW) + 30 LP Fachdidaktik (FD)

Sem.	Module	FW	FD
1	CS 110 Praktische Informatik I (Programmierung)	9	
2	CS 210 Praktische Informatik II (Algorithmen & Datenstrukturen) <sup>(1)</sup>	5	4
4	CS 410 Datenbanksysteme <sup>(1)</sup>	4	4
3	CS 310 Praktische Informatik III (Konzepte von Programmiersprachen)	4	4
4 oder 6	CS 460 Theoretische Informatik <sup>(1)</sup>	6	3
1 oder 3	CS 180 Mathematik I (Lineare Algebra) [CS 380 Logik/Diskrete Math.] <sup>(2)</sup>	9	
4 oder 6	CS 505 Didaktik der Informatik		3
	<b>Summe Pflichtveranstaltungen</b>	<b>37</b>	<b>18</b>
5 bis 7	<b>Wahlpflichtbereich Fachwissenschaften:</b> Drei Wahlpflichtmodule <sup>(3)</sup> mit jeweils 5-9 LP aus zwei verschiedenen Bereichen Systementwicklung, Systemumgebung oder Theoretische Informatik	20	
7 bis 8	CS 600 Seminar aus den Bereichen Systementwicklung, Systemumgebung oder Theoretische Informatik	3	
6 oder 8	<b>Wahlpflichtbereich Fachdidaktik:</b> CS 601 Fachdidaktik Vertiefung (3 LP)		3
5 bis 8	CS 604 Proseminar (4 LP), CS 605 Tutorenseminar <sup>(4)</sup> (4 LP), CS 420 Software-Praktikum <sup>(5)</sup> oder CS 606 Gesellschaftliche Aspekte (4 LP)		4
6 oder 8	CS 602 Schulpraktische Studien II (oder CS 603 Kompensationsmodul hierzu)		5
	<b>Summe Wahlpflichtbereich</b>	<b>23</b>	<b>12</b>
	<b>Gesamtsumme</b>	<b>60</b>	<b>30</b>

### Erläuterungen:

(1) Die Pflichtmodule

- CS 210 Praktische Informatik II (Algorithmen und Datenstrukturen)
- CS 310 Praktische Informatik III (Konzepte von Programmiersprachen)
- CS 410 Datenbanksysteme
- CS 460 Theoretische Informatik

werden als **Schnittstellenmodule** angeboten, in denen fachwissenschaftliche und fachdidaktische Anteile in einem Modul organisiert sind. Zu den Vorlesungen werden jeweils fachdidaktische Übungen oder ein fachdidaktisches Begleitseminar angeboten.

(2) Studierende, die **Mathematik als zweites Fach** studieren, besuchen statt der CS 180 Mathematik I im ersten Semester die CS 380 Logik/Diskrete Mathematik im dritten Semester.

(3) Höchstens zwei der Wahlpflichtmodule dürfen Grundmodule (CS 140 Technische Informatik I, CS 240 Technische Informatik II, CS 340 Einführung in die Softwaretechnik oder CS 420 Softwarepraktikum) sein.

- (4) Das CS 605 **Tutorenseminar** vertieft fachdidaktische Inhalte der Schnittstellenmodule. Die Teilnehmer können als studentische Hilfskräfte in den entsprechenden Lehrveranstaltungen der Schnittstellenmodule eingesetzt werden.
- (5) Das CS 420 **Software-Praktikum** ist ein Schnittstellenmodul, das mit vier Leistungspunkten im Wahlpflichtbereich Fachdidaktik und mit vier Leistungspunkten im Wahlpflichtbereich Fachwissenschaften (Bereich Systementwicklung) eingebracht werden kann. Es setzt die Vorlesung „Einführung in die Softwaretechnik“ voraus. Im Praktikum bilden die Lehramtsstudierenden eine eigene Gruppe mit Schwerpunktsetzung auf fachdidaktischen Themen.

**Regelstudienpläne**

Die folgenden Studienpläne geben Empfehlungen für den Studienablauf. Die „Didaktik der Informatik“ wird nur alle zwei Jahre im Sommersemester angeboten, muss also im vierten oder sechsten Fachsemester belegt werden. Im Gegenzug kann die Theoretische Informatik im vierten oder sechsten Fachsemester absolviert werden. Im ersten Plan wird angenommen, dass die „Didaktik der Informatik“ im vierten Fachsemester belegt wird. Im zweiten Plan wird sie im sechsten Fachsemester gehört. Studierende mit Mathematik als zweites Fach belegen im dritten Fachsemester das Modul Logik/Diskrete Mathematik, während alle anderen im ersten oder dritten Fachsemester das Modul Mathematik I absolvieren sollten.

**Plan A: Einführung in die Fachdidaktik im 4. Fachsemester**

Sem	Systementw.	Systemumbg.	Theorie/Math.	Fachdidaktik	Leist. punkte
1	Praktische I		Mathematik 1		18
2	Praktische II				9
3			Praktische III (Logik/DiMa)		8 (+9)
4		Datenbanksys		Didaktik der Inf	11
5	.....	2 Wahlpflichtmodule .....			13
6			Theor. Inf.	Vertiefung	12
7	.....	Wahlpflichtmodul .....		Schulpr. Stud. II / Kompensationsm.	12
8	.....	Seminar.....		Wahlpflicht	7

**Plan B: Einführung in die Fachdidaktik im 6. Fachsemester**

Sem	Systementw.	Systemumbg.	Theorie/Math.	Fachdidaktik	Punkte
1	Praktische I		Mathematik I		18
2	Praktische II				9
3			Praktische III (Logik/DiMa)		8 (+9)
4		Datenbanksys	Theor. Inf.		17
5	.....	2 Wahlpflichtmodule.....			13
6	.....	Wahlpflichtmodul .....		Didaktik der Inf	10
7	.....	Seminar.....		Schulpr. Stud. II / Kompensationsm.	8
8				Vertiefung Wahlpflicht	7

**Zwischenprüfung und Prüfungsmodule für die Erste Staatsprüfung**

Für die Zuerkennung der Zwischenprüfung sind spätestens bis zum Ende des 4. Fachsemesters, in besonders begründeten Ausnahmen bis zum Ende des 6. Fachsemesters, mindestens 35 Leistungspunkte in den Modulen CS 110 „Praktische Informatik I“, CS 210 „Praktische Informatik II“, CS 180 „Mathematik I“ bzw. CS 380 „Logik/Diskrete Mathematik“ sowie in einem der Module CS 310 „Praktische Informatik III“ oder CS 410 „Datenbanksysteme“ oder CS 460 „Theoretische Informatik“ zu erwerben. Leistungspunkte gelten erst als erworben, wenn die Modulprüfung bestanden ist.

**Erste Staatsprüfung**

Bei der Bildung der Gesamtnote für die Erste Staatsprüfung (HLbG § 29, Abs.2 Ziff.1) werden die Noten aus den nachstehend aufgeführten Modulen berücksichtigt:

- Eines der Module „Praktische Informatik II“ oder „Datenbanksysteme“
- Eines der Module „Theoretische Informatik“ oder „Praktische Informatik III“
- Eines der drei Wahlpflichtmodule aus den Fachwissenschaften

Des Weiteren die Modulabschlussnoten der fachdidaktischen Module „Fachdidaktik Vertiefung“ und das gewählte fachdidaktische Schnittstellenmodul. Sofern zwei fachdidaktische Module aus dem anderen Fach der individuellen Studienkombination eingebracht werden, findet das fachdidaktische Modul „Fachdidaktik Vertiefung“ Berücksichtigung.

Sofern die/der Studierende keinen anders lautenden Antrag stellt, werden jeweils die notenbesten Module in die Gesamtnote der Ersten Staatsprüfung eingebracht.

Die erste Staatsprüfung erstreckt sich auf die beiden Wahlpflichtmodule aus c), die nicht im Sinne von § 29 HLbG in die Gesamtnote der Ersten Staatsprüfung eingebracht wurden.

## 11.3 Modulhandbuch

### 11.3.1 Fachwissenschaftliche Pflichtmodule

<b>Modulbezeichnung</b>	<b>CS 110: Praktische Informatik I - Einführung in die Programmierung</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Algorithmenbegriff</li> <li>• Kenntnisse der Techniken und Werkzeuge für die Programmentwicklung</li> <li>• Kenntnisse im Bereich der imperativen Programmierung</li> <li>• Grundlagen der Programmierung mit Rekursion</li> <li>• Grundlegende Kenntnisse im Bereich der objektorientierten Programmierung</li> <li>• Erlernen einer objektorientierten Programmiersprache</li> <li>• Umgang mit Software-Entwicklungsumgebungen</li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>• Charakterisierung von Algorithmen</li> <li>• Programmiersprache Java</li> <li>• Kontrollstrukturen</li> <li>• Datentypen und ihre Konstruktion</li> <li>• Rekursion</li> <li>• Objekte und Klassen</li> <li>• Klassenerweiterung, Vererbung und Polymorphie</li> <li>• Klassenbibliotheken</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben, Bestehen von Zwischentests gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Keine
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std, Selbststudium 180 Std
<b>Leistungspunkte</b>	9
<b>Art der Prüfungen</b>	Bestehen einer von zwei Abschlussklausuren Note: Die Note entspricht der Note der zuerst bestandenen Abschlussklausur.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester; jährlich im Wintersemester
<b>Verwendbarkeit des Moduls</b>	Pflichtmodul im Lehramts- Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 180: Mathematik I</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Erwerb von Basiswissen und Fertigkeiten in „Lineare Algebra“</li> <li>• Erkennen von Querverbindungen zur Informatik</li> <li>• Verständnis für grundlegende Prinzipien algebraischer und linearer Strukturen</li> <li>• Schulung des Abstraktionsvermögens</li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>• Mengentheoretische und algebraische Grundlagen <ul style="list-style-type: none"> <li>○ Elemente der Logik, Grundlagen der Mengenlehre, Abbildungen</li> <li>○ Gruppen, Rekursionen, Körper</li> </ul> </li> <li>• Vektorräume und lineare Abbildungen <ul style="list-style-type: none"> <li>○ Basis, Dimensionen, Quotientenräume</li> <li>○ Homomorphiesatz</li> </ul> </li> <li>• Matrizen und lineare Gleichungssysteme <ul style="list-style-type: none"> <li>○ Darstellung linearer Abbildungen, Basiswechsel</li> <li>○ Lösungsalgorithmen, Determinanten</li> </ul> </li> <li>• Unitäre Vektorräume</li> <li>• Skalarprodukte, Orthogonalität, Eigenwerte, Spektraltheorie</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 5 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben
<b>Teilnahmevoraussetzungen</b>	Keine
<b>Arbeitsaufwand</b>	Präsenzzeit 100 Std, Selbststudium 170 Std
<b>Leistungspunkte</b>	9
<b>Art der Prüfungen</b>	Bestehen einer von zwei Abschlussklausuren Note: Die Note entspricht der Note der zuerst bestandenen Abschlussklausuren.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, jährlich im Wintersemester
<b>Verwendbarkeit des Moduls</b>	Pflichtmodul im Lehramtsstudiengang Informatik, falls als zweites Fach nicht Mathematik studiert wird; Exportmodul für andere Studiengänge



<b>Modulbezeichnung</b>	<b>CS 380: Logik und Diskrete Mathematik</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Vermittlung grundlegender Fertigkeiten im Umgang mit logischen Formeln, insbesondere im Modellieren und Formalisieren</li> <li>• Einsicht in grundlegende Fragen des Beweisens und die Möglichkeiten und Grenzen des Automatisierens von Beweisen</li> <li>• Verständnis grundlegender Begriffe der diskreten Mathematik</li> <li>• Umgang mit für die Informatik relevanten diskreten Strukturen</li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>• Aussagenlogik <ul style="list-style-type: none"> <li>○ Syntax und Semantik, äquivalente Umformungen und Normalformen</li> <li>○ Erfüllbarkeitstests, Beweiskalküle, Vollständigkeit</li> </ul> </li> <li>• Prädikatenlogik <ul style="list-style-type: none"> <li>○ Syntax und Semantik, äquivalente Umformungen und Normalformen</li> <li>○ Beweiskalküle, Hinweis auf Vollständigkeitssatz</li> </ul> </li> <li>• Elementare Kombinatorik <ul style="list-style-type: none"> <li>○ kombinatorische Strukturen (Mengen, Permutationen)</li> <li>○ grundlegende Abzählmethoden (Binomialkoeffizient, einfache erzeugende Funktionen)</li> </ul> </li> <li>• Graphentheorie <ul style="list-style-type: none"> <li>○ einfache Grapheigenschaften (Zusammenhang, Azyklizität, Färbbarkeit)</li> <li>○ kombinatorische Algorithmen</li> </ul> </li> <li>• Codierung <ul style="list-style-type: none"> <li>○ präfix-freie Codes</li> <li>○ einfache Eigenschaften linearer Codes</li> </ul> </li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse in Linearer Algebra und Analysis
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std, Selbststudium 180 Std
<b>Leistungspunkte</b>	9
<b>Art der Prüfungen</b>	Bestehen einer von zwei Abschlussklausuren Note: Die Note entspricht der Note der zuerst bestandenen Abschlussklausur
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, jährlich im Wintersemester
<b>Verwendbarkeit des Moduls</b>	Pflichtmodul im Lehramtsstudiengang Informatik, falls Mathematik als weiteres Fach studiert wird; Exportmodul für andere Studiengänge

## 11.3.2 Schnittstellenmodule

<b>Modulbezeichnung</b>	<b>CS 210: Praktische Informatik II mit fachdidaktischen Übungen - Datenstrukturen und Algorithmen</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Verstehen von Analysetechniken für Algorithmen und Datenstrukturen</li> <li>• Anwendung von Datenstrukturen und Algorithmen zur Lösung elementarer Problemen</li> <li>• Entwurf von Algorithmen und Datenstrukturen</li> <li>• Vertiefung der Programmierkenntnisse</li> <li>• Einordnung unterschiedlicher Konzepte und Beurteilung der Einsatzmöglichkeit im Schulunterricht</li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>• Analyse von Algorithmen und Datenstrukturen <ul style="list-style-type: none"> <li>◦ Komplexität von Algorithmen, Asymptotische Analyse</li> </ul> </li> <li>• Elementare Datenstrukturen <ul style="list-style-type: none"> <li>◦ Listen, Stacks, Queues</li> <li>◦ Suchbäume - Hashtabellen - Graphen</li> </ul> </li> <li>• Elementare Algorithmen <ul style="list-style-type: none"> <li>◦ Suchen - Sortieren - Datenkompression</li> </ul> </li> <li>• Entwurfsprinzipien effizienter Algorithmen</li> <li>• Didaktische Aufbereitung des Stoffes für den Schulunterricht</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS, fachdidaktisches Begleitseminar (bei geringer Zahl an Lehramtsstudierenden) Erfolgreiche Bearbeitung und Präsentation von Übungsaufgaben, Kurzvorträge in den Übungen zu schulbezogenen Themen der Vorlesung im Hinblick auf die Umsetzung in der gymnasialen Oberstufe, Bestehen von Zwischentests gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus dem Modul Praktische Informatik I
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std, Selbststudium 180 Std
<b>Leistungspunkte</b>	9, davon 4 im Bereich Fachdidaktik
<b>Art der Prüfungen</b>	Bestehen einer von zwei Abschlussklausuren. Noten: Die Note entspricht der Note der zuerst bestandenen Abschlussklausur.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester; jährlich im Sommersemester
<b>Verwendbarkeit des Moduls</b>	Schnittstellenmodul im Lehramtsstudiengang Informatik

<b>Modulbezeichnung</b>	<b>CS 310: Praktische Informatik III - Konzepte von Programmiersprachen mit fachdidaktischen Übungen</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Beurteilung der Korrektheit von Algorithmen und Datentypen, Durchführung von Korrektheitsbeweisen</li> <li>• Erlernen einer deklarativen Programmiersprache</li> <li>• Erkennen und Anwendung von Abstraktion bei der Programmentwicklung</li> <li>• Verstehen und Erkennen von sprachübergreifenden Konzepten; Unterscheidung verschiedener Programmierparadigmen und ihrer Anwendungsbereiche</li> <li>• Beschreibung der Semantik von Programmiersprachen</li> <li>• Einordnung der Konzepte und Beurteilung der Einsatzmöglichkeit im Schulunterricht</li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>• Deklarative Programmierung <ul style="list-style-type: none"> <li>▪ Funktionale Programmierung: Rekursive Funktionsgleichungen, Algebraische Datenstrukturen und Pattern Matching, Funktionen höherer Ordnung, Polymorphe Typsysteme, Typinferenz</li> <li>▪ oder Logik-Programmierung: Hornklauselprogramme, Unifikation und Resolution, Backtracking, definite clause grammars, Differenzlisten</li> </ul> </li> <li>• Methoden der Beschreibung der Semantik von Programmiersprachen: operationell, denotationell, axiomatisch</li> <li>• Verifikation: Klasseninvarianten, Schleifeninvarianten, Verifikationsbedingungen, Hoare-Kalkül, Korrektheitsbeweise</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS, fachdidaktisches Begleitseminar (bei geringer Zahl an Lehramtsstudierenden) Erfolgreiche Bearbeitung von Übungsaufgaben, aktive Mitarbeit im Tutorium, Bestehen von Zwischentests gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse von Programmiersprachen wie sie in der Vorlesung Praktische Informatik I vermittelt werden
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std, Selbststudium 150 Std
<b>Leistungspunkte</b>	8, davon 4 im Bereich Fachdidaktik
<b>Art der Prüfungen</b>	Bestehen einer von zwei Abschlussklausuren. Note: Note der zuerst bestandenen Abschlussklausur
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, jährlich im Wintersemester
<b>Verwendbarkeit des Moduls</b>	Schnittstellenmodul im Lehramtsstudiengang Informatik

<b>Modulbezeichnung</b>	<b>CS 460: Theoretische Informatik mit fachdidaktischen Übungen</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Umgang mit regulären Ausdrücken, endlichen Automaten und</li> <li>• Grammatiken. Erkennen der Möglichkeiten und Grenzen</li> <li>• Verständnis formaler Modelle des Berechnens.</li> <li>• Prinzipielle Grenzen des algorithmischen Rechnens</li> <li>• Grenzen des effizienten Lösens von Problemen.</li> <li>• Einordnung der Konzepte und Beurteilung der Einsatzmöglichkeit im Schulunterricht</li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>• Reguläre Sprachen: <ul style="list-style-type: none"> <li>○ reguläre Ausdrücke, endliche Automaten, Äquivalenz</li> <li>○ Anwendung: Lexikalische Analyse</li> </ul> </li> <li>• Kontextfreie Sprachen: <ul style="list-style-type: none"> <li>○ Kontextfreie Grammatiken und Sprachen.</li> <li>○ Allgemeinere Grammatik-Typen</li> <li>○ Anwendung: Syntax-Analyse</li> </ul> </li> <li>• Berechenbarkeit: <ul style="list-style-type: none"> <li>○ Modelle der Berechenbarkeit</li> <li>○ Äquivalenz, Church-Turing-These;</li> <li>○ Entscheidbarkeit, unlösbare Probleme</li> </ul> </li> <li>• Komplexitätstheorie: <ul style="list-style-type: none"> <li>○ Aufwand von Berechnungen; P und NP;</li> <li>○ Reduktionen und NP-vollständige Probleme</li> </ul> </li> <li>• Didaktische Aufbereitung des Stoffes für den Schulunterricht</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS, fachdidaktisches Begleitseminar (bei geringer Zahl an Lehramtsstudierenden) Erfolgreiche Bearbeitung und Präsentation von Übungsaufgaben, Kurzvorträge in den Übungen zu schulbezogenen Themen der Vorlesung im Hinblick auf die Umsetzung in der gymnasialen Oberstufe, Bestehen von Zwischentests gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Mathematische Grundkenntnisse wie sie im Modul Mathematik I vermittelt werden.
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std, Selbststudium 180 Std
<b>Leistungspunkte</b>	9, davon 3 im Bereich Fachdidaktik
<b>Art der Prüfungen</b>	Bestehen einer von zwei Abschlussklausuren. Note: Note der zuerst bestandenen Abschlussklausur
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, jährlich im Sommersemester
<b>Verwendbarkeit des Moduls</b>	Schnittstellenmodul im Lehramtsstudiengang Informatik

<b>Modulbezeichnung</b>	<b>CS 410: Datenbanksysteme mit fachdidaktischen Übungen</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Datenmodellierung</li> <li>• Umsetzung von Datenmodellen in einen Datenbankentwurf</li> <li>• Verstehen wichtiger Anfragekalküle</li> <li>• Kenntnisse über die Grundfunktionalität von SQL</li> <li>• Grundlegende Kenntnisse zu Transaktionen</li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>• Methoden und Werkzeuge für Datenmodellierung</li> <li>• Datenbankmodelle</li> <li>• Anfragesprachen</li> <li>• Anwendungsprogrammierung</li> <li>• Integritätsbedingungen</li> <li>• Anfragebearbeitung</li> <li>• Datenbankentwurf</li> <li>• Transaktionen</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS, fachdidaktisches Begleitseminar (bei geringer Zahl an Lehramtsstudierenden) Erfolgreiche Bearbeitung von Übungsaufgaben, Bestehen von Zwischentests gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Kenntnisse wie sie in der Praktischen Informatik I und II vermittelt werden
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std , Selbststudium 150 Std
<b>Leistungspunkte</b>	8, davon 4 im Bereich Fachdidaktik
<b>Art der Prüfungen</b>	Bestehen einer von zwei Abschlussklausuren. Note: Die Note entspricht der Note der zuerst bestandenen Abschlussklausur.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, jährlich im Sommersemester
<b>Verwendbarkeit des Moduls</b>	Schnittstellenmodul im Lehramtsstudiengang Informatik

## 11.3.3 Fachdidaktikmodule

<b>Modulbezeichnung</b>	<b>CS 505 – Einführung in die Didaktik der Informatik</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• didaktische Aufbereitung fachwissenschaftlicher Inhalte</li> <li>• Einsatz spezifischer Lehr-/Lernformen für die Wissensvermittlung</li> <li>• Einschätzung und Vermittlung der Stellung des Unterrichtsgegenstandes in der Fachwissenschaft, im interdisziplinären Zusammenhang und in der Alltagswelt</li> <li>• Planung von Rechneinsatz an Schulen unter pädagogischen Gesichtspunkten</li> </ul>
<b>Thema und Inhalt</b>	<ol style="list-style-type: none"> <li>1. Allgemeinbildung – Informatikunterricht</li> <li>2. Didaktik – Fachdidaktik</li> <li>3. Unterrichtsplanung (lang-, mittel-, kurzfristig) - didaktische Analyse - Lehrplan</li> <li>4. Computer im Unterricht</li> <li>5. Informations- und kommunikationstechnische Grundbildung (IKG)</li> <li>6. Leistungsmessung und -bewertung</li> <li>7. „Projekte“ im Informatikunterricht</li> <li>8. Beispiele für den Unterricht: <ol style="list-style-type: none"> <li>a. „Einführung in HTML" als Thema für den IKG-Unterricht</li> <li>b. „LOGO in der Sekundarstufe I"</li> <li>c. „Grafikprogrammierung (Java-Applets) zur Einführung in das Programmieren"</li> <li>d. Turtle-Grafik als "Strang" im Informatik-Unterricht</li> <li>e. „Datenbanksysteme“</li> <li>f. "Einsatzmöglichkeiten der Scriptsprache PHP in der Oberstufe"</li> </ol> </li> </ol>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung oder Proseminar 2 SWS Erfolgreiche Ausarbeitung eines Themas, Vortragspräsentation und schriftliche Ausgestaltung im Seminar gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Vorlesungen Praktische Informatik I und Mathematik I/Lineare Algebra
<b>Arbeitsaufwand</b>	Präsenzzeit 30 Std., Selbststudium 60 Std.
<b>Leistungspunkte</b>	3
<b>Art der Prüfungen</b>	Abschlussprüfung Note: Die Note ergibt sich aus der Note der Modulprüfung
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, jedes zweite Sommersemester
<b>Verwendbarkeit des Moduls</b>	Pflichtmodul im Lehramtsstudiengang Informatik

### 11.3.4 Fachwissenschaftliche Wahlpflichtmodule und Schnittstellenmodule

- **Bereich Systementwicklung**

<b>Modulbezeichnung</b>	<b>CS 340: Einführung in die Softwaretechnik</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Entwicklung von Software mit Werkzeugen aus der Softwaretechnik</li> <li>• Fähigkeiten zur Analyse und Modellierung von Problemstellungen aus Anwendungsbereichen</li> <li>• Kenntnisse von Datenmodellen und Modellierungssprachen</li> <li>• Kenntnisse über Entwurfsprinzipien, Modularisierung und Software-Architektur</li> <li>• Kenntnisse über Test- und Integrationsverfahren von Software</li> <li>• Grundkenntnisse über Projekt-Management und Qualitätssicherung</li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>• Grundlagen und Terminologie der Softwaretechnik</li> <li>• System- und Anforderungsanalyse</li> <li>• Fachlicher Entwurf, System- und Datenmodellierung,</li> <li>• Entwurfsprinzipien, Modularisierung, Software-Architektur</li> <li>• Software-Test und -Integration</li> <li>• Projekt-Management und Qualitätssicherung</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 3 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben, Bestehen von Zwischentests gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus der Praktischen Informatik I und II
<b>Arbeitsaufwand</b>	Präsenzzeit 75 Std., Selbststudium 135 Std.
<b>Leistungspunkte</b>	7
<b>Art der Prüfungen</b>	Bestehen einer von zwei Abschlussklausuren. Note: Die Note entspricht der Note der zuerst bestandenen Abschlussklausur
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, jährlich im Wintersemester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang, Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 420: Software-Praktikum mit fachdidaktischer Ausrichtung</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Bearbeitung einer größeren Software-Entwicklungsaufgabe durch alle Projektphasen unter besonderer Berücksichtigung schulrelevanter Themen und im Hinblick auf eine schulische Umsetzung</li> <li>• Vertiefung der Programmierkenntnisse, Erproben der Arbeit im Team und Strukturierung des Projekts unter Anleitung nach Prinzipien der Softwaretechnik.</li> <li>• Darstellen und Präsentieren von Arbeits- und Projektergebnissen</li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>• Problemanalyse, Entwurf, Implementierung, Test und Integration eines größeren Softwaresystems.</li> <li>• Qualitätssicherung und Dokumentation des Systems</li> <li>• Präsentation von Zwischen- und Endergebnissen.</li> <li>• Diskussion und Planung der Durchführung von Software-Projekten in der gymnasialen Oberstufe</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Einführungsvorlesung 4 Std, Übungen 4 SWS, Praktikum
<b>Teilnahmevoraussetzungen</b>	Erfolgreicher Abschluss des Moduls Praktische Informatik I, Erwartet werden darüber hinaus Grundkenntnisse aus den Vorlesungen Praktische Informatik II und Einführung in die Softwaretechnik
<b>Arbeitsaufwand</b>	Präsenzzeit 60 Std., Selbststudium 180 Std.
<b>Leistungspunkte</b>	8
<b>Art der Prüfungen</b>	(Gemeinsame) Auslieferung des erstellten Systems. Dokumentation und Präsentation von Ergebnissen. Note: Keine Benotung (bestanden / nicht bestanden)
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, jährlich im Sommersemester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang



<b>Modulbezeichnung</b>	<b>CS 501 – Programmieren in C++</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Grundlagen von Programmiersprachen</li> <li>• Objektorientierte Programmierung</li> <li>• C und C++</li> <li>• Programmieren mit C++</li> </ul>
<b>Thema und Inhalt</b>	<ol style="list-style-type: none"> <li>1. Elementare C/C++ Programme</li> <li>2. Ausdrücke, Zuweisungen</li> <li>3. Anweisungen: bedingte Anweisungen; Schleifen</li> <li>4. Datenstrukturen</li> <li>5. Unterprogramme, Funktionen, Rekursion</li> <li>6. Überladen von Funktionen; Template Funktionen</li> <li>7. Modularisierung, Header Dateien</li> <li>8. Klassen: Konstruktoren, Destruktoren</li> <li>9. Überladen von Operatoren</li> <li>10. Datei Bearbeitung</li> <li>11. Klassen: Vererbung, Polymorphie</li> <li>12. Template Klassen</li> </ol>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS und Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Kenntnisse aus den Vorlesungen Praktische Informatik I – II
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std , Selbststudium 150 Std
<b>Leistungspunkte</b>	8
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur / eines Kolloquiums Note: Die Note entspricht der Klausur- bzw. Kolloquiumsnote.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, in der Regel jedes Wintersemester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 507 – Moderne Methoden der Systementwicklung</b>
<b>Kompetenzen und Qualifikationsziel</b>	Ziel des Moduls ist die Vermittlung neuer Forschungsergebnisse aus aktuellen wissenschaftlichen Beiträgen in nationalen und internationalen Fachzeitschriften sowie von Konferenzen.
<b>Thema und Inhalt</b>	Es werden Inhalte der neueren Forschung aus dem Bereich Systementwicklung behandelt. Die spezielle fachliche Ausrichtung des Moduls wird vom Dozenten festgelegt.
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 3 SWS und Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben, gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Kenntnisse aus den Vorlesungen Praktische Informatik I – II
<b>Arbeitsaufwand</b>	Präsenzzeit 60 Std , Selbststudium 150 Std
<b>Leistungspunkte</b>	7
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder eines Kolloquiums Note: Die Note entspricht der Klausur- bzw. Kolloquiumsnote.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, unregelmäßig
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsgang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 531 - Systemanalyse und Modellierung</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Grundverständnis für Systemtheorie, Softwaretechnik, Requirements</li> <li>• Engineering und Theorie der Modellierung</li> <li>• Kenntnisse von Modellierungssprachen, insb. UML</li> <li>• Fähigkeiten zur Analyse und Modellierung von Software-Anwendungen</li> </ul>
<b>Thema und Inhalt</b>	<ol style="list-style-type: none"> <li>1. Grundlagen: Systemtheorie, Softwaretechnik, Requirements Engineering, Modelle und Modellierung</li> <li>2. Analyseverfahren: Strukturierte Analyse, Objektorientierte Analyse, Anwendungsfall-Analyse (Use case analysis),</li> <li>3. Verfahren und Beschreibungsmittel zur Anwendungsmodellierung: Entity- Relationship-Modell, Informations-/Funktionsstrukturanalyse, Unified Modeling Language (UML), Klassen- und Objektdiagramme, Aktivitäts-, Zustands- und Sequenzdiagramme, Metamodelle</li> <li>4. Werkzeuge: CASE, Data Dictionaries, UML-Werkzeuge</li> </ol> <p>Dieses Modul beginnt den Softwaretechnik-Zyklus und kann mit "Entwurf großer Software Systeme", aber auch mit Vorlesungen über Datenbanksysteme fortgesetzt werden.</p>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 3 SWS und Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Kenntnisse aus den Vorlesungen Praktische Informatik I – III und Einführung in die Softwaretechnik
<b>Arbeitsaufwand</b>	Präsenzzeit 75 Std , Selbststudium 135 Std
<b>Leistungspunkte</b>	7
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur / eines Kolloquiums Note: Die Note entspricht der Klausur- bzw. Kolloquiumsnote.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle drei bis vier Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 532 - Entwurf großer Softwaresysteme</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Grundverständnis für komplexe Software-Systeme und Entwicklungs- Prozesse</li> <li>• Kenntnisse im Verferten fachlicher und technischer Entwürfe</li> <li>• Theoretische und praktische Fertigkeiten bei der Modularisierung, Architektur- Entwicklung und beim Test von Software-Systemen</li> </ul>
<b>Thema und Inhalt</b>	<ol style="list-style-type: none"> <li>1. Software-Entwicklungs-Prozesse, Terminologie</li> <li>2. Grundlagen der Systemanalyse und Anwendungsmodellierung (Zusammenfassung)</li> <li>3. Prinzipien des Software-Entwurfs: Geheimnisprinzip, Daten-Abstraktion, Modularisierung, Objekt-orientierter Entwurf, Komponentenbildung</li> <li>4. Software-Architekturen (Schichtenmodelle, Komponenten, Frameworks, Entwurfsmuster), UML als Entwurfssprache, EIFFEL</li> <li>5. Software-Realisierung: Implementierung, Test und Integration; Inkrementelle und evolutionäre Systementwicklung, Agile Entwicklungsmethoden</li> <li>6. Software-Entwurfs-Werkzeuge</li> </ol>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 3 SWS und Praktikum 2 SWS Erfolgreiche Bearbeitung von Teilaufgaben im begleitenden Software-Praktikum, Bestehen von Zwischenkolloquien gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Kenntnisse aus den Vorlesungen Praktische Informatik I – III und Einführung in die Softwaretechnik
<b>Arbeitsaufwand</b>	Präsenzzeit 75 Std , Selbststudium 135 Std
<b>Leistungspunkte</b>	7
<b>Art der Prüfungen</b>	Bestehen des Abschluss-Kolloquiums. Note: Die Note ergibt sich aus der Note der Modulprüfung.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle drei bis vier Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 533 - Software-Projektmanagement und Qualitätssicherung</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Grundverständnis für das Management komplexer Software-Entwicklungs- vorhaben, der beteiligten Rollen, Aktivitäten und der dabei entstehenden Ergebnisse</li> <li>• Kenntnisse über Ziele, Methoden und Verfahren der Software- Qualitätssicherung</li> <li>• Verständnis für die Bedeutung des menschlichen Faktors bei der Software- Entwicklung (Führungsstile, Motivation, Kommunikation im Projekt, Benutzerbeteiligung, dynamisches Management)</li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>• Grundlagen: Vorgehensmodelle, Software-Lebenszyklus, Prozessmodelle</li> <li>• Anforderungen und Aufgaben des Projekt-Managements, Projektplanung, -steuerung und -abschluss</li> <li>• Aufwandsschätzung für Software-Projekte</li> <li>• Erfahrungsberichte und Fallstudien von Software-Projekten</li> <li>• Ausgewählte Prozessmodelle: Rational Unified Process (RUP), EOS-Modell für objektorientierte und evolutionäre SW-Entwicklung</li> <li>• Qualitätsbegriff, Qualitäts-Merkmale, Qualitätssicherung (QS), Konstruktive und analytische QS-Maßnahmen</li> <li>• Der menschliche Faktor im Software-Prozess</li> <li>• Software-Verwaltung und Konfigurations-Management</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 3 SWS, Praktikum 2 SWS Erfolgreiche Bearbeitung von Teilaufgaben im begleitenden Software-Praktikum, Bestehen von Zwischenkolloquien gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Vorlesungen Praktische Informatik I – III und Einführung in die Softwaretechnik. Dieses Modul setzt den Softwaretechnik-Zyklus fort. Kenntnisse aus den Modulen "Systemanalyse und Anwendungs-Modellierung" und "Entwurf großer Software-Systeme" sind hilfreich, aber nicht unbedingt Voraussetzung für den Besuch dieses Moduls.
<b>Arbeitsaufwand</b>	Präsenzzeit 75 Std., Selbststudium 135 Std.
<b>Leistungspunkte</b>	7
<b>Art der Prüfungen</b>	Bestehen des Abschluss-Kolloquiums. Note: Die Note ergibt sich aus der Note der Modulprüfung
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle drei bis vier Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 634 - Modellierung von Informationssystemen und Wissensrepräsentation</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Grundverständnis für die Theorie der Modellierung und ihre Bedeutung für die Softwaretechnik</li> <li>• Vertiefende Kenntnisse in Modellierungsmethoden und -sprachen, insb. UML</li> <li>• Umgang mit Methoden und Werkzeugen zur konzeptionellen Modellierung, Wissensrepräsentation und Ontologie-Entwicklung</li> </ul>
<b>Thema und Inhalt</b>	<ol style="list-style-type: none"> <li>1. Begriffliche Grundlagen: System- und Modellbegriff, Zeichenprozesse und Semiotik, Information und Wissen, Informationssysteme, Klassen von Informationssystemen</li> <li>2. Grundlagen der Modellierung, Modellierungskonzepte, traditionelle und objektorientierte Modellierungstechniken</li> <li>3. Analyse und Modellierung mit der UML, Analysemuster, neue Forschungsansätze zur Modellierung</li> <li>4. Einsatz von Formaler Begriffsanalyse (FBA) zur Modellierung</li> <li>5. FRISCO: Ein begrifflicher Rahmen für die Informationssystem-Entwicklung</li> <li>6. Ontologien und Techniken zur Wissensrepräsentation: Semantische Netze, Begriffsgraphen, Ontologiesprachen</li> </ol>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 2 SWS, Seminar 1 SWS Erfolgreiche Teilnahme am Begleitseminar, selbständige Bearbeitung eines Seminarthemas, Halten eines Seminarvortrags gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Vorlesungen Praktische Informatik I – III sowie mindestens einer Veranstaltung aus dem Softwaretechnik-Zyklus
<b>Arbeitsaufwand</b>	Präsenzzeit 45 Std., Selbststudium 105 Std.
<b>Leistungspunkte</b>	5
<b>Art der Prüfungen</b>	Abfassen einer Seminararbeit. Note: Die Note ergibt sich aus der Note der Modulprüfung
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle vier Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 551 – Grundlagen des Compilerbaus</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Aspekte von Programmiersprachen (Lexik, Syntax, Semantik, Pragmatik)</li> <li>• Aufbau von Compilern, Compilerphasen, Bootstrapping</li> <li>• Theoretische Basis für Analysen</li> <li>• Analysewerkzeuge zu Lexik und Syntax verstehen, erstellen und anwenden</li> <li>• Grundprinzipien der Codeerzeugung und der denotationellen Semantik, abstrakte Maschinen</li> <li>• Programmieren von Compilerteilen</li> </ul>
<b>Thema und Inhalt</b>	<p>Compiler übersetzen Programme aus höheren Programmiersprachen in ausführbaren Maschinencode. In der Vorlesung werden Konzepte und Methoden, die für die verschiedenen Phasen von Compilern entwickelt wurden, vor- und gegenübergestellt. Im einzelnen werden folgende Themen behandelt:</p> <ol style="list-style-type: none"> <li>1. Lexikalische Analyse: reguläre Ausdrücke, endliche Automaten, flm-Prinzip, Scannergeneratoren</li> <li>2. Syntaktische Analyse: Kontextfreie Grammatiken, RD-Parsing mit LL(k)- Grammatiken, SR-Parsing mit LR(k)-Grammatiken, Präzedenz-gesteuerte Parser, Parsergeneratoren</li> <li>3. Semantische Analyse: Attributgrammatiken und ihre Berechnung</li> <li>4. Zwischencode-Erzeugung</li> <li>5. Code-Optimierung</li> </ol> <p>In den begleitenden Übungen werden Compiler Teile in Haskell (siehe <a href="http://www.haskell.org">www.haskell.org</a>) programmiert und Haskell-Werkzeuge zur Compilerentwicklung eingesetzt.</p>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben,gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus dem Modul Theoretische Informatik
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std., Selbststudium 180 Std.
<b>Leistungspunkte</b>	9
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder eines Kolloquiums Note: Die Note entspricht der Note der Abschlussprüfung.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle zwei bis drei Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik (Bereiche Theoretische Informatik und Systementwicklung); Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 553 – Parallele Programmierung</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Erlernen von Techniken zur parallelen Programmierung</li> <li>• Erstellung von parallelen Programmen</li> <li>• Gegenüberstellung verschiedener paralleler Paradigma</li> <li>• Bewertung von parallelen Programmen</li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>• Modelle der Parallelverarbeitung</li> <li>• Die nebenläufige Programmiersprache MPD</li> <li>• Das Message Passing Interface MPI</li> <li>• Datenparallelität und High Performance FORTRAN (HPF)</li> <li>• Parallele Algorithmen</li> <li>• Alternative Ansätze zur parallelen Programmierung</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben,gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus dem Modul Praktische Informatik I
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std., Selbststudium 180 Std.
<b>Leistungspunkte</b>	9
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder eines Abschlusskolloquiums Note: Note der Abschlussprüfung
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle drei bis vier Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge



<b>Modulbezeichnung</b>	<b>CS 561 – Modellgetriebene Softwareentwicklung</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>○ Überblick über Softwareentwicklungsprozesse</li> <li>○ Verständnis des Paradigmas der modellgetriebenen Softwareentwicklung</li> <li>○ Modellierung von Geschäftsanwendungen</li> <li>○ Anwendung der modellgetriebenen Softwareentwicklung auf Geschäftsanwendungen</li> <li>○ Verständnis für die Möglichkeiten und Grenzen der modellgetriebenen Softwareentwicklung</li> <li>○ Grundverständnis für die in der LV vorgestellten Werkzeuge, durch Übungen vertieft</li> </ul>
<b>Thema und Inhalt</b>	<p>Durch die modellgetriebene Softwareentwicklung, die zunehmend an Bedeutung gewinnt, können aus verschiedenen Arten von visuellen Modellen lauffähige Programme generiert werden. Von dieser Vorgehensweise verspricht man sich eine erhebliche Steigerung der Entwicklungsgeschwindigkeit, weil die Software auf einer höheren Ebene entwickelt werden kann. Durch die Abstraktion von Implementierungsdetails erhofft man sich auch eine bessere Beherrschung der Komplexität von Anwendungen. Außerdem soll die Qualität der erstellten Software gesteigert werden. Zur Modellierung von Softwaresystemen haben sich verschiedene visuelle Sprachen etabliert. Die bekannteste ist zur Zeit die Unified Modeling Language (UML). Die Generatoren erzeugen domänen- und plattformspezifische Lösungen aus gegebenen Modellen.</p> <p>Im Rahmen der Lehrveranstaltung beginnen wir mit einer Einführung in die modellgetriebene Softwareentwicklung anhand des Eclipse Modeling Frameworks (EMF). Darauf aufbauend betrachten wir die modellgetriebene Entwicklung von visuellen Editoren für domänenspezifische Modellierungssprachen auf der Basis weiterer Eclipse Plugins, wie z.B. das Eclipse Graphical Modeling Framework (GMF). Als dritten Schwerpunkt behandeln wir die modellgetriebene Entwicklung von Geschäftsanwendungen basierend auf Java 2 Enterprise Edition (J2EE). Hier verwenden wir den Generator AndroMDA.</p>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben, gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Modulen Praktische Informatik I-III
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std., Selbststudium 180 Std.
<b>Leistungspunkte</b>	9
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder eines Abschlusskolloquiums Note: Note der Abschlussprüfung
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle drei bis vier Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik (Bereiche Theoretische Informatik und Systementwicklung); Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 562 – Visuelle Sprachen</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>○ Verständnis für die Unterschiede zwischen textuellen und visuellen Sprachen</li> <li>○ Erlernen verschiedener Methoden zur Definition visueller Sprachen</li> <li>○ Grundkenntnisse im Umgang mit Designerwerkzeugen für visuelle Sprachen</li> <li>○ Einarbeitung in eine unbekannte visuelle Sprache und ihre Bewertung anhand einer kleinen Testapplikation</li> </ul>
<b>Thema und Inhalt</b>	<p>Zur Softwareentwicklung werden vermehrt Sprachen eingesetzt, die dem Entwickler erlauben, die zu entwickelnde Software visuell zu entwerfen, zu implementieren, zu testen und zu verwalten. Hierzu gehören Modellierungssprachen wie z.B. die Unified Modeling Language (<i>UML</i>), aber auch <i>SDL</i>, <i>Petrinetze</i> und <i>Statecharts</i>. Nicht ganz so bekannt sind visuelle Programmiersprachen, wie z.B. <i>ProGraph</i> und <i>StageCast</i>. Visuelle Entwicklungsumgebungen wie z.B. <i>Delphi</i>, <i>JBuilder</i>, <i>Eclipse</i>, enthalten Komponenten für das visuelle Verwalten von Softwareprojekten, für die visuelle Implementierung von graphischen Benutzeroberflächen als auch für Softwarevisualisierungen. Im Rahmen der Lehrveranstaltung werden wir verschiedene konkrete visuelle Sprachen und Entwicklungsumgebungen genauer betrachten und die Vor- und Nachteile hinsichtlich ihrer visuellen Konzepte diskutieren. Dabei werden die Möglichkeiten und Grenzen visueller Modellierung im Softwareentwicklungsprozess aufgezeigt.</p>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 2 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben, gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Modulen Praktische Informatik I-III
<b>Arbeitsaufwand</b>	Präsenzzeit 60 Std., Selbststudium 120 Std.
<b>Leistungspunkte</b>	6
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder eines Abschlusskolloquiums Note: Note der Abschlussprüfung
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle drei bis vier Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik (Bereiche Theoretische Informatik und Systementwicklung); Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 607 – Moderne Methoden der Systementwicklung II</b>
<b>Kompetenzen und Qualifikationsziel</b>	Ziel des Moduls ist die Vermittlung fortgeschrittener neuer Forschungsergebnisse aus aktuellen wissenschaftlichen Beiträgen in nationalen und internationalen Fachzeitschriften sowie von Konferenzen.
<b>Thema und Inhalt</b>	Es werden Inhalte der neueren Forschung aus dem Bereich Systementwicklung behandelt. Die spezielle fachliche Ausrichtung des Moduls wird vom Dozenten festgelegt.
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 3 SWS und Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben, gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Kenntnisse aus den Vorlesungen Praktische Informatik I – II
<b>Arbeitsaufwand</b>	Präsenzzeit 60 Std , Selbststudium 150 Std
<b>Leistungspunkte</b>	7
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder eines Kolloquiums Note: Die Note entspricht der Klausur- bzw. Kolloquiumsnote.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, unregelmäßig
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 652 – Parallele und verteilte Algorithmen</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Erlernen und Einordnung verschiedener Grundmuster paralleler Verarbeitung</li> <li>• Gegenüberstellung verschiedener Verfahren zur parallelen Problemlösung</li> <li>• Erstellen von parallelen Programmen</li> </ul>
<b>Thema und Inhalt</b>	<p>Nach einer Einführung in die Grundbegriffe der Parallelverarbeitung werden zunächst elementare parallele Algorithmen diskutiert. Anschließend werden parallele Algorithmen für verschiedene Problemklassen wie Sortieren, Matrizen-Operationen, Graphenverfahren behandelt. Außerdem werden verteilte Basisverfahren wie Schnappschussverfahren, Terminationserkennung, Garbage Collection und Verfahren für verteilte Probleme vorgestellt.</p> <p>In den begleitenden Übungen sollen verschiedene Verfahren in C / MPI (PVM) und in Eden (paralleles Haskell) implementiert werden.</p>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 3 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben, gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus dem Modul Praktische Informatik II
<b>Arbeitsaufwand</b>	Präsenzzeit 75 Std., Selbststudium 165 Std.
<b>Leistungspunkte</b>	8
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder eines Abschlusskolloquiums Note: Note der Abschlussprüfung
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle drei bis vier Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik (Bereiche Theoretische Informatik und Systementwicklung); Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 653 – Parallelität in funktionalen Programmiersprachen</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Erlernen verschiedener Methoden, Parallelität in funktionalen Sprachen zu behandeln und auszudrücken</li> <li>• Erstellung paralleler funktionaler Programme in verschiedenen Sprachen</li> </ul>
<b>Thema und Inhalt</b>	<p>Nach einer Einführung in die verschiedenen Möglichkeiten, funktionale Sprachen um parallele Konzepte zu erweitern, werden die wichtigsten Methoden im Detail besprochen: automatische Parallelisierung, semi-explizite und explizite Sprachen, Datenparallelität, algorithmische Skelette und Kostenmodelle.</p> <p>In den Übungen wird die parallele funktionale Sprache Eden eingesetzt. Eden erweitert Haskell um Konstrukte zur expliziten Definition und Verwaltung von Prozessen.</p>
<b>Organisations-, Lehr- und Lernform</b>	<p>Vorlesung 3 SWS, Übungen 2 SWS  Erfolgreiche Bearbeitung von Übungsaufgaben, gem.  Modulankündigung</p>
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus dem Modul Praktische Informatik III
<b>Arbeitsaufwand</b>	Präsenzzeit 75 Std., Selbststudium 165 Std.
<b>Leistungspunkte</b>	8
<b>Art der Prüfungen</b>	<p>Bestehen einer Abschlussklausur oder eines Abschlusskolloquiums  Note: Note der Abschlussprüfung</p>
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle drei bis vier Semester
<b>Verwendbarkeit des Moduls</b>	<p>Wahlpflichtmodul im Lehramtsstudiengang Informatik (Bereiche Theoretische Informatik und Systementwicklung);  Exportmodul für andere Studiengänge</p>

<b>Modulbezeichnung</b>	<b>CS 661 – Softwarequalität</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Erlernen möglicher Verfahren zur Qualitätsprüfung und -verbesserung von Software</li> <li>• Erwerb von Grundkenntnissen in den vorgestellten Techniken</li> </ul>
<b>Thema und Inhalt</b>	Ein wichtiges Thema in der Softwareentwicklung ist die Qualität von Software. Typische Qualitätskriterien sind die Korrektheit, Robustheit, Änderbarkeit, Wiederverwendbarkeit und Benutzerfreundlichkeit von Software. In der Lehrveranstaltung betrachten wir verschiedene Techniken, wie z.B. Softwremetriken, Refactoring und Testverfahren, um die Software nach syntaktischen und semantischen Gesichtspunkten zu untersuchen und zu verbessern.
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 2 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben
<b>Teilnahmevoraussetzungen</b>	Grundkenntnisse aus den Modulen Praktische Informatik I-III
<b>Arbeitsaufwand</b>	Präsenzzeit 60 Std., Selbststudium 120 Std.
<b>Leistungspunkte</b>	6
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder eines Abschlusskolloquiums Note: Note der Abschlussprüfung
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle drei bis vier Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik (Bereiche Theoretische Informatik und Systementwicklung); Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 581 – Grafikprogrammierung 1</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Grundlagen der 2D Grafik</li> <li>• Grafikpakete</li> <li>• Algorithmen der 2D Grafik</li> </ul>
<b>Thema und Inhalt</b>	<ol style="list-style-type: none"> <li>1. Graphik - Hardware</li> <li>2. VektorGrafik / Rastergrafik</li> <li>3. Algorithmen zur 2D Graphik</li> <li>4. 2D Graphik Programmierung</li> <li>5. Fraktale, L-Systeme, ...</li> <li>6. Computational Geometry</li> <li>7. Splines und Beziere</li> <li>8. Dateiformate, Bildverarbeitung</li> <li>9. Farben</li> </ol>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben und bestehen mündlicher Zwischenprüfungen, Durchführung von Semesterprojekten gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Vorlesungen Praktische Informatik I und II
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std., Selbststudium 180 Std.
<b>Leistungspunkte</b>	9
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur. Note: Note der Abschlussklausur
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, in der Regel jedes zweite Wintersemester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 681 – Grafikprogrammierung 2</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Grundlagen der 3D Grafik</li> <li>• Beleuchtungsmodelle,</li> <li>• Sichtbarkeit</li> <li>• Ray Tracing und Radiosity</li> </ul>
<b>Thema und Inhalt</b>	<ol style="list-style-type: none"> <li>1. Einführung in die 3D-Grafik</li> <li>2. Mathematische Grundlagen</li> <li>3. Repräsentation von Objekten: Modellierung</li> <li>4. Einführung in OpenGL</li> <li>5. Beleuchtungsmodelle</li> <li>6. Sichtbarkeits-Algorithmen</li> <li>7. Schatten und Texturen</li> <li>8. Interaktion und Animation</li> <li>9. VRML und X3D</li> <li>10. Globale Beleuchtungsmodelle: <ul style="list-style-type: none"> <li>• -&gt; Ray Tracing</li> <li>• -&gt; Radiosity</li> </ul> </li> <li>11. Animation</li> </ol>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben und bestehen mündlicher Zwischenprüfungen, Durchführung von Semesterprojekten gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Vorlesungen Praktische Informatik I und II
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std., Selbststudium 180 Std.
<b>Leistungspunkte</b>	9
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur. Note: Note der Abschlussklausur
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, in der Regel alle vier Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge



- **Bereich Systemumgebung**

<b>Modulbezeichnung</b>	<b>CS 140 – Technische Informatik I (Rechnerstrukturen)</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Grundlegende Kenntnisse digitaler Schaltungen</li> <li>• Einführung in die Boolesche Algebra</li> <li>• Grundlegende Kenntnisse der Rechnerarithmetik</li> <li>• Grundlegende Kenntnisse des Aufbaus einer CPU</li> <li>• Einführung in die Assemblerprogrammierung</li> <li>• Überblick über Rechnerarchitekturkonzepte</li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>• Technologische Grundlagen</li> <li>• Schaltnetze und Schaltwerke</li> <li>• Boolesche Algebra</li> <li>• Rechnerarithmetik</li> <li>• Aufbau und Organisation einer CPU</li> <li>• Assemblerprogrammierung</li> <li>• Rechnerarchitekturen</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben und bestehen von Zwischentests gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Keine
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std., Selbststudium 180 Std.
<b>Leistungspunkte</b>	9
<b>Art der Prüfungen</b>	Bestehen von einer von zwei Abschlussklausuren Note: Die Note entspricht der Note der zuerst bestandenen Abschlussklausur.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, in jedem Wintersemester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 240 – Techn. Inf. II (Betriebssysteme, Rechnerkomm.)</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Grundlegende Kenntnisse der Konzepte von Betriebssystemen</li> <li>• Umgang mit Unix-Betriebssystemkommandos</li> <li>• Grundlegende Kenntnisse der Methoden der Rechnerkommunikation</li> <li>• Grundlegende Kenntnisse der Konzepte des Internets</li> </ul>
<b>Thema und Inhalt</b>	<p>I. Grundlagen von Betriebssystemen</p> <ul style="list-style-type: none"> <li>• Prozesse, Betriebsmittelverwaltung</li> <li>• Verklemmungen</li> <li>• Speicherverwaltung, Dateisysteme</li> </ul> <p>II. Unix-Einführung</p> <p>III. Grundlagen der Rechnerkommunikation</p> <ul style="list-style-type: none"> <li>• Protokolle: ISO-OSI, TCP/IP</li> <li>• Leitungen: Twisted Pair, Koax, Glasfaser</li> <li>• Bitcodierungen</li> <li>• Serielle Schnittstellen, parallele Schnittstellen, MODEMs, ISDN</li> <li>• Lokale Netze LANs, WANs, GANs, MANs</li> <li>• Ethernet, TokenRing, Bridges, Router, FDDI, ATM</li> </ul> <p>IV. Das Internet</p> <ul style="list-style-type: none"> <li>• Die TCP/IP Protokolle im Einzelnen</li> <li>• Internet Adressen, Struktur, Dienste</li> <li>• Internet: Basisdienste, mittlere Dienste, höhere Protokolle</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben und bestehen von Zwischentests gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Keine
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std., Selbststudium 180 Std.
<b>Leistungspunkte</b>	9
<b>Art der Prüfungen</b>	Bestehen einer von zwei Abschlussklausuren Note: Die Note entspricht der Note der zuerst bestandenen Abschlussklausur
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, in jedem Sommersemester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 508 – Aktuelle Methoden zur Informationsverarbeitung</b>
<b>Kompetenzen und Qualifikationsziel</b>	Ziel des Moduls ist die Vermittlung neuer Forschungsergebnisse aus aktuellen wissenschaftlichen Beiträgen in nationalen und internationalen Fachzeitschriften sowie von Konferenzen.
<b>Thema und Inhalt</b>	Es werden Inhalte der neueren Forschung aus den Bereichen Verteilte Systeme, Multimedia, Informationssysteme, Künstliche Intelligenz, Bioinformatik und weiterer interdisziplinärer Bezüge behandelt. Die spezielle fachliche Ausrichtung des Moduls wird vom Dozenten festgelegt.
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 3 SWS und Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Kenntnisse aus den Vorlesungen Praktische Informatik I – II
<b>Arbeitsaufwand</b>	Präsenzzeit 60 Std , Selbststudium 150 Std
<b>Leistungspunkte</b>	7
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder eines Kolloquiums Note: Die Note entspricht der Klausur- bzw. Kolloquiumsnote.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, unregelmäßig
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 511 – Rechnernetze</b>
<b>Kompetenzen und Qualifikationsziel</b>	Die Vorlesung "Rechnernetze" beschäftigt sich mit Aufbau und Funktionsweise von Rechnernetzen. Lernziel des Moduls ist es, die Hörer in die Lage zu versetzen, das Verhalten von Netzwerken (inklusive des Internets) in groben Zügen analysieren und verstehen zu können. Ein besonderer Schwerpunkt liegt dabei auf den Problemstellungen und der Funktionalität von Netzwerksoftware. Nach dem Besuch des Moduls sollten die Hörer in der Lage sein, nach zusätzlicher Einarbeitung in die jeweiligen Netzwerkkumgebungen, eigene Netzwerk-Module zu entwickeln.
<b>Thema und Inhalt</b>	<ol style="list-style-type: none"> <li>1. Einleitung</li> <li>2. Anwendungsschicht</li> <li>3. Transportschicht</li> <li>4. Netzwerkschicht</li> <li>5. Verbindungsschicht</li> <li>6. Physikalische Schicht</li> <li>7. Netzwerksicherheit</li> <li>8. Multimedianeetze</li> <li>9. Netzwerkmanagement</li> </ol>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Vorlesungen Praktische Informatik I und II sowie Technische Informatik II
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std., Selbststudium 180 Std.
<b>Leistungspunkte</b>	9
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder eines Kolloquiums Note: Die Note entspricht der Note der Abschlussprüfung.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, in der Regel jedes zweite Wintersemester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 512 – Betriebssysteme</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Erlernen der Funktionsweise von Betriebssystemen</li> <li>• Möglichkeit der eigenständigen Programmierung von Betriebssystemmodulen</li> </ul>
<b>Thema und Inhalt</b>	<ol style="list-style-type: none"> <li>1. Einleitung</li> <li>2. Hardware-Grundlagen</li> <li>3. Prozess/Thread-Verwaltung <ul style="list-style-type: none"> <li>• Synchronisation</li> <li>• Scheduling</li> <li>• Verklemmungen</li> </ul> </li> <li>4. Speicherverwaltung</li> <li>5. Ein-/Ausgabe-Verwaltung</li> <li>6. Dateisysteme</li> <li>7. Sicherheit und Schutz</li> <li>8. Fallstudien</li> </ol>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 2 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Vorlesungen Praktische Informatik I und II sowie Technische Informatik II
<b>Arbeitsaufwand</b>	Präsenzzeit 60 Std., Selbststudium 120 Std.
<b>Leistungspunkte</b>	6
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder eines Kolloquiums Note: Die Note entspricht der Note der Abschlussprüfung.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, in der Regel jedes zweite Wintersemester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 513 – Verteilte Systeme</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Erlernen der Funktionsweise verteilter Systeme</li> <li>• Algorithmen für Problemstellungen verteilter Systeme</li> </ul>
<b>Thema und Inhalt</b>	<ol style="list-style-type: none"> <li>1. Einführung</li> <li>2. Strukturen verteilter Systeme</li> <li>3. Kommunikation</li> <li>4. Synchronisation</li> <li>5. Prozessverwaltung</li> <li>6. Speicherverwaltung</li> <li>7. Namensgebung</li> <li>8. Verteilte Dateisysteme</li> <li>9. Fehlertoleranz</li> <li>10. Sicherheit</li> <li>11. Middleware</li> <li>12. Fallbeispiele</li> </ol>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 2 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Vorlesungen Praktische Informatik I und II, Technische Informatik II, Betriebssysteme
<b>Arbeitsaufwand</b>	Präsenzzeit 60 Std., Selbststudium 120 Std.
<b>Leistungspunkte</b>	6
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder eines Kolloquiums Note: Die Note entspricht der Note der Abschlussprüfung.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, in der Regel jedes 2. Sommersemester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 514 – Grid Computing</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Erlernen der Grundkonzepte von Grid Computing</li> <li>• Erstellung von Grid-Software</li> <li>• Entwicklung verteilter Grid-Anwendungen</li> </ul>
<b>Thema und Inhalt</b>	<p>Als Grid bezeichnet man ein Internet-basiertes System, welches das Anbieten, Auffinden und die Nutzung von Ressourcen über verschiedene administrative Domänen hinweg ermöglicht. Grid Computing beruht auf dem Konzept der koordinierten gemeinsamen Nutzung von Rechnern, Software und Daten sowie der Lösung von Problemen in dynamischen, mehrere Institutionen umfassenden virtuellen Organisationen. Solche kooperativen Problemlösungsumgebungen werden in zunehmendem Maße in vielen Bereichen von Industrie und Wissenschaft benötigt. Dabei werden u.a. Faktoren wie Verfügbarkeit, Authentifizierung, Kosten-Abrechnung und Quality-of-Service als Nutzer-Anforderungen berücksichtigt. Als Analogie werden hier die Eigenschaften des Stromnetzes ("Power Grid") gesehen - die jeweilige Ressource ist im Idealfall unabhängig von der Quelle gleichbleibend verfügbar.</p> <p>In der Vorlesung werden zunächst die grundlegenden Konzepte des Grid Computing vorgestellt, dann einige der in den letzten Jahren entwickelten Softwareinfrastrukturen und Anwendungen. Darauf aufbauend wird die "Open Grid Service Architecture (OGSA)" als Grundlage des service-orientierten Ansatzes zum Grid Computing vorgestellt. Weiterhin werden Programmiermodelle, Softwarewerkzeuge und Middleware für die Erstellung, Ausführung und Verwaltung service-orientierter Anwendungen erläutert. Ein Schwerpunkt des Moduls liegt auf der praktischen Erprobung der vermittelten Konzepte in Form von interaktiven Programmierübungen.</p>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 2 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Vorlesungen Praktische Informatik I und II, Technische Informatik II, Betriebssysteme, Internet-Middleware
<b>Arbeitsaufwand</b>	Präsenzzeit 60 Std., Selbststudium 120 Std.
<b>Leistungspunkte</b>	6
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder eines Kolloquiums Note: Die Note entspricht der Note der Abschlussprüfung.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, unregelmäßig
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik, Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 541 – Einführung in die Bioinformatik</b>
<b>Kompetenzen und Qualifikationsziel</b>	<p>Im Rahmen des Moduls sollen die Studierenden</p> <ul style="list-style-type: none"> <li>• die Bedeutung der Bioinformatik für die modernen Biowissenschaften erkennen;</li> <li>• mit den speziellen Eigenschaften molekularbiologischer Daten vertraut werden;</li> <li>• wichtige Problemklassen wie Sequenzanalyse, Strukturanalyse, Expressionsanalyse und phylogenetische Analyse kennen lernen;</li> <li>• die wichtigsten algorithmischen und methodischen Grundlagen der Bioinformatik, insbesondere der Sequenzanalyse, erlernen;</li> <li>• Informatik-Methoden selbständig auf molekularbiologische Fragestellungen anwenden.</li> </ul>
<b>Thema und Inhalt</b>	<p>Unter Bioinformatik versteht man die Entwicklung und Nutzung von Methoden und Werkzeugen der Informatik für Fragen der Biologie, wobei der Molekularbiologie in neuerer Zeit besondere Bedeutung zukommt. Durch die breite Verfügbarkeit genomischer Daten ist die Biologie von einer phänomenologischen, beschreibenden zu einer analytischen, erklärenden Wissenschaft geworden. Im Zusammenhang mit der Gewinnung, Verwaltung, Analyse und Interpretation genomischer Daten spielen Informatik-Methoden inzwischen eine unverzichtbare Rolle.</p> <p>Inhalt:</p> <ul style="list-style-type: none"> <li>• Grundlagen und Bedeutung der Bioinformatik</li> <li>• Molekularbiologische Grundlagen</li> <li>• Paarweises und multiples Alignment von Sequenzen mittels dynamischer Programmierung</li> <li>• Heuristische Verfahren des Sequenzvergleichs</li> <li>• Methoden der phylogenetischen Analyse</li> <li>• Hidden-Markov-Modelle und deren Anwendungen</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	<p>Vorlesung 4 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung</p>
<b>Teilnahmevoraussetzungen</b>	<p>Keine Grundkenntnisse in praktischer Informatik werden erwartet. Die wichtigsten molekularbiologischen Grundlagen werden im Rahmen der Vorlesung rekapituliert, entsprechende Vorkenntnisse daher <b>nicht</b> vorausgesetzt. Grundkenntnisse in Wahrscheinlichkeitsrechnung sind hilfreich aber nicht notwendig.</p>
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std., Selbststudium 180 Std.
<b>Leistungspunkte</b>	9
<b>Art der Prüfungen</b>	Bestehen einer Abschlussprüfung (Klausur oder Kolloquium) Note: Note der Abschlussprüfung
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle 4 Semester
<b>Verwendbarkeit des</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik,



<b>Moduls</b>	Exportmodul für andere Studiengänge
<b>Modulbezeichnung</b>	<b>CS 542 – Maschinelles Lernen</b>
<b>Kompetenzen und Qualifikationsziel</b>	<p>Im Laufe des Moduls sollen die Studierenden</p> <ul style="list-style-type: none"> <li>• grundlegende Fragestellungen und Ziele des maschinellen Lernens verstehen</li> <li>• mit speziellen Problemklassen, wie dem überwachten Lernen (Klassifikation und Regression), vertraut werden</li> <li>• sich wichtige Methoden des maschinellen Lernens erarbeiten</li> <li>• mit Konzepten zur Evaluierung von Lernverfahren vertraut werden</li> <li>• in die Lage versetzt werden, praktische Problemstellungen mit Verfahren des maschinellen Lernens eigenständig zu lösen.</li> </ul>
<b>Thema und Inhalt</b>	<p>Methoden des maschinellen Lernens sowie angrenzender Gebiete wie Wissensentdeckung und Data Mining sind zentraler Gegenstand der aktuellen Forschung im Bereich intelligenter Systeme und werden bereits in einer Vielzahl praktischer Anwendungen eingesetzt.</p> <p>Inhalt:</p> <ul style="list-style-type: none"> <li>• Einführung und grundlegende Konzepte</li> <li>• Begriffslernen und Versionenräume</li> <li>• Fallbasiertes Lernen</li> <li>• Entscheidungsbäume</li> <li>• Regellernen</li> <li>• Bayessche Inferenz</li> <li>• Support Vector Machines</li> <li>• Erweiterungen und Meta-Techniken</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Grundvorlesungen in praktischer Informatik werden erwartet.
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std., Selbststudium 180 Std.
<b>Leistungspunkte</b>	9
<b>Art der Prüfungen</b>	Bestehen einer Abschlussprüfung (Klausur oder Kolloquium) Note: Note der Abschlussprüfung
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig jedes 4. Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik, Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 543 – Computational Intelligence</b>
<b>Kompetenzen und Qualifikationsziel</b>	<p>Im Rahmen der Vorlesung sollen die Studierenden</p> <ul style="list-style-type: none"> <li>• die Paradigmen der "Computational Intelligence" (CI) und deren Bedeutung für die Entwicklung intelligenter Systeme verstehen</li> <li>• das Potential von CI-Methoden für praktische Anwendungen einschätzen können</li> <li>• Grundlagen wichtiger CI-Methoden beherrschen und auf praktische Fragestellungen anwenden können</li> <li>• die besondere Bedeutung von hybriden Ansätzen erkennen.</li> </ul>
<b>Thema und Inhalt</b>	<p>Unter dem Begriff "Computational Intelligence" (CI) fasst man ein Ensemble von Methoden zusammen, dessen Eckpfeiler gebildet werden durch die Paradigmen der konnektionistischen Informationsverarbeitung (künstliche neuronale Netze), der evolutionären Algorithmen sowie der Fuzzy-Logik. Auf Grundlage dieser biologisch motivierten und durch die menschliche Informationsverarbeitung inspirierten Paradigmen wurden in den letzten Jahren viel versprechende Methoden entwickelt, die sich als hervorragende Werkzeuge zum Entwurf intelligenter Systeme etabliert haben und Konzepte der klassischen künstlichen Intelligenz sinnvoll ergänzen. Hierbei haben sich hybride Ansätze, die Vorteile aus zwei oder mehreren der genannten Bereiche kombinieren, als besonders erfolgreich erwiesen.</p> <p>Inhalt:</p> <ol style="list-style-type: none"> <li>1. Einführung in die CI</li> <li>2. Fuzzy Sets: Grundlagen</li> <li>3. Fuzzy Sets: Anwendungen</li> <li>4. Rough Sets: Grundlagen</li> <li>5. Rough Sets: Anwendungen</li> <li>6. Evolutionäre Algorithmen: Grundlagen</li> <li>7. Evolutionäre Algorithmen: Anwendungen</li> <li>8. Hybride Systeme</li> </ol>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Grundvorlesungen in praktischer Informatik werden erwartet.
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std., Selbststudium 180 Std.
<b>Leistungspunkte</b>	9
<b>Art der Prüfungen</b>	Bestehen einer Abschlussprüfung (Klausur oder Kolloquium) Note: Note der Abschlussprüfung
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle 4 Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik, Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 544 – Fuzzy Systeme</b>
<b>Kompetenzen und Qualifikationsziel</b>	<p>Im Rahmen der Vorlesung sollen die Studierenden</p> <ul style="list-style-type: none"> <li>• den Nutzen von Konzepten und Prinzipien der mehrwertigen Logik in intelligenten und wissensbasierten Systemen erkennen;</li> <li>• theoretische Grundlagen der Theorie unscharfer Mengen erlernen;</li> <li>• mit Anwendungen von Fuzzy-Methoden in unterschiedlichen Bereichen vertraut werden und entsprechende Methoden selbständig anwenden.</li> </ul>
<b>Thema und Inhalt</b>	<p>Intelligente Systeme müssen in der Lage sein, unsichere, unvollständige oder unpräzise Information zu verarbeiten. Methoden auf der Grundlage der Theorie unscharfer Mengen (Fuzzy Sets) spielen in diesem Zusammenhang eine wichtige Rolle.</p> <p>Inhalt:</p> <ul style="list-style-type: none"> <li>• Einführung und Motivation</li> <li>• Mengenoperationen</li> <li>• Verallgemeinerte logische Operatoren</li> <li>• Fuzzy-Relationen und deren Anwendungen</li> <li>• Fuzzy-Regeln und Approximatives Schließen</li> <li>• Hybride Systeme</li> <li>• Datenbasierte Fuzzy-Modellierung</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 2 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Grundvorlesungen in praktischer Informatik werden erwartet.
<b>Arbeitsaufwand</b>	Präsenzzeit 60 Std., Selbststudium 120 Std.
<b>Leistungspunkte</b>	6
<b>Art der Prüfungen</b>	Bestehen einer Abschlussprüfung (Klausur oder Kolloquium) Note: Note der Abschlussprüfung
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle 4 Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik, Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 571 – Index- und Speicherstrukturen</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Erlernen von Techniken zur Entwicklung externer Datenstrukturen und Algorithmen</li> <li>• Bewertung externer Datenstrukturen bezgl. Laufzeit, Speicherplatz und Durchsatz</li> <li>• Erstellung von Datenstrukturen innerhalb einer systemnahen Umgebung</li> <li>• Umgang mit Indexstrukturen in kommerziellen Datenbanksystemen</li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>• Dateisysteme</li> <li>• B-Bäume und Varianten</li> <li>• Externe Hashverfahren</li> <li>• Multidimensionale Indexstrukturen</li> <li>• Indexstrukturen für Geo-Datenbanken</li> <li>• Zeit-Indexstrukturen</li> <li>• Hochdimensionale Indexstrukturen</li> <li>• Indexstrukturen in kommerziellen Datenbanksystemen</li> <li>• Lock-Protokolle und Recovery für Indexstrukturen</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 2 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben, bestehen von Zwischenklausuren gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Vorlesungen Praktische Informatik II und Datenbanksysteme
<b>Arbeitsaufwand</b>	Präsenzzeit 60 Std., Selbststudium 120 Std.
<b>Leistungspunkte</b>	6
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur. Note: Note der Abschlussklausur
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle 3-4 Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik, Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 572 – Information Retrieval</b>
<b>Kompetenzen und Qualifikationsziel</b>	Studierende sollen Aufgabenstellung, Modelle und Methoden des Information Retrieval kennen lernen. Dabei soll die Fähigkeit zur Nutzung und zur Mitwirkung bei der Konzeption von Suchmaschinen für Internet- und Intranet-Applikationen vermittelt werden. Ebenso sollen die grundsätzlichen Implementierungstechniken und ihre Vor- und Nachteile verstanden werden.
<b>Thema und Inhalt</b>	Gegenstände des Information Retrieval (IR) ist die Suche nach Dokumenten. Traditionell handelt es sich dabei um Textdokumente. In neuerer Zeit kommt verstärkt die Suche nach multimedialen Dokumenten (Bilder, Audio, Video, Hypertext-Dokumente) hinzu. Das Moduls betrachtet die wesentlichen Modelle des Information Retrieval und Algorithmen zu ihrer Umsetzung. Auch Fragen der Evaluierung von IR-Systemen werden betrachtet. Folgende Bereiche werden behandelt: Motivation und Einführung, Evaluierung von IR-Systemen, Berücksichtigung der Vagheit in Sprache, einfache IR-Modelle und ihre Implementierung, das Vektorraummodell, Formate zur Dokumenten-Wissensverwaltung, Alternativen zur globalen Suche, Multimedia Information Retrieval, Suchmaschinen im World Wide Web.
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 2 SWS, Übungen 2 SWS
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Vorlesungen Praktische Informatik I und II sowie Datenbanksysteme
<b>Arbeitsaufwand</b>	Präsenzzeit 60 Std., Selbststudium 120 Std.
<b>Leistungspunkte</b>	6
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur. Note: Note der Abschlussklausur
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle 4 Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik, Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 573 – Geo-Datenbanken</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Erwerb von Kenntnissen zur Erweiterung objektrelationaler Datenbanksysteme für Geo-Anwendungen</li> <li>• Prinzipien grundlegender Algorithmen und Datenstrukturen für Geo- Datenbanken</li> <li>• Kenntnisse über die Datenmodelle für Geo-Daten</li> <li>• Anfrageverarbeitung in Geo-Datenbanken</li> <li>• Umgang mit kommerziellen Geo-Informationssystemen</li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>• Einführung in erweiterbare Datenbanksysteme</li> <li>• Modellierung von Geo-Daten</li> <li>• Anfrageverarbeitung in Geo-Datenbanken</li> <li>• Indexierung von Geo-Datenbanken</li> <li>• Kartenüberdeckung</li> <li>• Algorithmen der Computational Geometry</li> <li>• Kommerzielle Geo-Informationssysteme</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 2 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Vorlesungen Praktische Informatik II und Datenbanksysteme
<b>Arbeitsaufwand</b>	Präsenzzeit 60 Std., Selbststudium 120 Std.
<b>Leistungspunkte</b>	6
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur. Note: Note der Abschlussklausur
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle 4 Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik, Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 591 – Knowledge Discovery</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• wissenschaftliche Vorgehensweise bei der Untersuchung von Datensammlungen mit dem Ziel neues und bislang unbekanntes Wissen zu entdecken</li> <li>• praktische Verwendung von explorativen statistischen Methoden zur Beschreibung und Analyse der Daten</li> <li>• Methoden der Visualisierung und Projektion von hochdimensionalen Daten</li> <li>• unterschiedliche Verfahren zur Clusterung von Daten und ihre Eigenheiten</li> <li>• Konstruktion und Analyse von Klassifikatoren</li> <li>• maschinelle Verfahren zu Gewinnung von Wissen</li> <li>• Wissensarten und ihre Nutzung in Expertensystemen</li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>• Gewinnung von Verteilungs- und Zusammenhangshypothesen</li> <li>• Regularisierung von Verteilungen</li> <li>• Entkoppelung von Korrelationen</li> <li>• Ähnlichkeitsdefinitionen für mehrdimensionale Datensätze</li> <li>• wissenschaftliche Visualisierung</li> <li>• Projektionsmethoden</li> <li>• Clusteralgorithmen und ihre Eigenschaften</li> <li>• Konstruktion von Klassifikatoren</li> <li>• Extraktion von Wissen (Maschinelles Lernen / Modellierung)</li> <li>• Datenbionische Verfahren (Selbstorganisation, Künstliches Leben)</li> <li>• Validierung der Einzelschritte des Knowledge Discovery.</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 3 SWS; Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben, aktive Teilnahme an den Übungen, Bestehen von Zwischentests gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Grundkenntnisse aus den Vorlesungen Praktische Informatik I – III
<b>Arbeitsaufwand</b>	Präsenzzeit 75 Std., Selbststudium 135 Std.
<b>Leistungspunkte</b>	7
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder eines Kolloquiums Note: Note der Abschlussprüfung
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, in jedem Wintersemester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 592 – Einführung in die Künstliche Intelligenz</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Fertigkeiten und Kenntnis der wichtigsten KI-Methoden und deren Anwendung in der Praxis.</li> <li>• Programmierung wissensbasierter Inferenzsysteme, in Prädikatenlogik (Prolog).</li> <li>• Wissensrepräsentationsformen</li> <li>• Problemlösungs-, Such- und Planungsalgorithmen.</li> <li>• Überblick über gebräuchliche Methoden des Schätzen: Bayes, Demster/Shafer, Fuzzy Inferenz.</li> <li>• Methoden des Wissenserwerbs: maschinelles Lernen, Knowledge Engineering</li> <li>• Einführung in nichtklassische Logiken</li> </ul>
<b>Thema und Inhalt</b>	<ol style="list-style-type: none"> <li>1. Programmieren in Prolog/ Prädikatenlogik /Constraints</li> <li>2. Wissen, Wissensrepräsentation, Inferenz</li> <li>3. Struktur wissensbasierter Systeme</li> <li>4. Wahrscheinlichkeitsbasiertes Schließen</li> <li>5. DS und Fuzzy Inferenz</li> <li>6. Knowledge Engineering und maschinelles Lernen</li> <li>7. Nichtklassische Logiken</li> <li>8. Praxis der wissensbasierten Systeme /Agentensysteme</li> </ol>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 2 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben und aktive Teilnahme an den Übungen gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Vorlesungen Praktische Informatik I - III
<b>Arbeitsaufwand</b>	Präsenzzeit 60 Std., Selbststudium 120 Std.
<b>Leistungspunkte</b>	6
<b>Art der Prüfungen</b>	Erfolgreiches Absolvieren eines Kolloquiums oder einer Klausur Note: Note der Klausur oder des Kolloquiums
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, jedes zweite Wintersemester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge



<b>Modulbezeichnung</b>	<b>CS 593 – Neuronale Netze</b>
<b>Kompetenzen und Qualifikationsziel</b>	Einführung in die Theorie der neuronalen Netze und Überblick über die verschiedenen Architekturen, Möglichkeiten und Grenzen künstlicher neuronaler Netze. Neben den gebräuchlichen überwacht lernenden Netzen wird insbesondere auf die unüberwacht lernenden neuronalen Netze eingegangen und das Paradigma der Selbstorganisation aufgezeigt. Ausgehend von einer konkreten Problemstellung sollen die Studierenden in der Lage sein eine datengetriebene Lösung für künstliche Neuronale Netze, unter Verwendung von vorgegebenen Programmen, zu entwerfen.
<b>Thema und Inhalt</b>	<ol style="list-style-type: none"> <li>1. Biologische neuronalen Netze</li> <li>2. Überwachte Lernverfahren</li> <li>3. Unüberwachte Lernverfahren</li> <li>4. Theoretische Analyse Neuronaler Netze</li> <li>5. Selbstorganisation und Emergenz</li> <li>6. Experimentdesign und Analyse</li> <li>7. Möglichkeiten und Grenzen der Modelle</li> </ol>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 2 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben und aktive Teilnahme an den Übungen gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Vorlesungen Praktische Informatik I - III
<b>Arbeitsaufwand</b>	Präsenzzeit 60 Std., Selbststudium 120 Std.
<b>Leistungspunkte</b>	6
<b>Art der Prüfungen</b>	Erfolgreiches Absolvieren eines Kolloquiums oder einer Klausur Note: Note der Klausur oder des Kolloquiums
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, jedes zweite Wintersemester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 608 – Aktuelle Methoden zur Informationsverarbeitung II</b>
<b>Kompetenzen und Qualifikationsziel</b>	Ziel des Moduls ist die Vermittlung fortgeschrittener neuer Forschungsergebnisse aus aktuellen wissenschaftlichen Beiträgen in nationalen und internationalen Fachzeitschriften sowie von Konferenzen.
<b>Thema und Inhalt</b>	Es werden Inhalte der neueren Forschung aus den Bereichen Verteilte Systeme, Multimedia, Informationssysteme, Künstliche Intelligenz, Bioinformatik und weiterer interdisziplinärer Bezüge behandelt. Die spezielle fachliche Ausrichtung des Moduls wird vom Dozenten festgelegt.
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 3 SWS und Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Kenntnisse aus den Vorlesungen Praktische Informatik I – II
<b>Arbeitsaufwand</b>	Präsenzzeit 60 Std , Selbststudium 150 Std
<b>Leistungspunkte</b>	7
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder eines Kolloquiums Note: Die Note entspricht der Klausur- bzw. Kolloquiumsnote.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, unregelmäßig
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 671 – Datenintegration</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Erlernen von Techniken zur Kopplung von Datenbanken</li> <li>• Erwerb von Kenntnissen zu Techniken bei der Schematransformation</li> <li>• Prinzipien des Datawarehousing</li> <li>• Analysetechniken für große Datenbanken</li> <li>• Prinzipien des Message Queuing</li> <li>• Grundlegende Kenntnisse zu Web-Datenbanken</li> </ul>
<b>Thema und Inhalt</b>	<p>Eines der zentralen Probleme der Informatik ist die Integration von Daten aus verschiedenen Datenbanken. In dieser Vorlesung werden hierfür die grundlegenden Techniken bei der Datenintegration vorgestellt.</p> <ul style="list-style-type: none"> <li>• Architektur für Data Warehouses</li> <li>• Extraktion von Daten und Metadaten</li> <li>• Datenvorverarbeitung</li> <li>• Techniken der Schematransformation</li> <li>• Schnelles Laden von Daten</li> <li>• Online Analyse im Datawarehouse</li> <li>• Kontinuierliches Laden und Datenströme (MessageQueuing)</li> <li>• Kopplungstechniken für Datenbanksysteme</li> <li>• Datenaustausch im Web</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 2 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus der Vorlesung Datenbanksysteme
<b>Arbeitsaufwand</b>	Präsenzzeit 60 Std., Selbststudium 120 Std.
<b>Leistungspunkte</b>	6
<b>Art der Prüfungen</b>	Bestehen einer mündlichen Prüfung. Note: Note der mündlichen Prüfung
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle 3-4 Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 682 – Multimediakommunikation</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Grundlagen von Multimedia bzw. Netzwerken</li> <li>• Repräsentation von Multimedia Informationen</li> <li>• Komprimierung</li> </ul>
<b>Thema und Inhalt</b>	<ol style="list-style-type: none"> <li>1. Multimedia Netzwerke, Multimedia Anwendungen</li> <li>2. Digitalisierung von Text, Audio, Bildern und Video</li> <li>3. Grundlagen der Komprimierung</li> <li>4. Text Komprimierung</li> <li>5. Audio Komprimierung, MP3, AAC, ...</li> <li>6. Bild Komprimierung, JPEG, JPEG2000</li> <li>7. Video Komprimierung allgemein</li> <li>8. Video Komprimierungs-Standards: H120, ...H264, MPEG1, ...MPEG4, DivX</li> </ol>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben; mündliche Zwischenprüfungen; Durchführen eines Video-Filmprojektes; gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Vorlesungen Technische Informatik II und Rechnernetze
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std., Selbststudium 180 Std.
<b>Leistungspunkte</b>	9
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur. Note: Note der Abschlussklausur
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, jedes zweite Sommersemester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 691 – Temporales Data Mining</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• wissenschaftliche Vorgehensweise bei der Untersuchung von Zeitreihen mit dem Ziel neue und bislang unbekannte zeitliche Muster zu entdecken</li> <li>• Kenntnis der wichtigsten Analyseverfahren wie Fourier- und Wavelet</li> <li>• Statistische Modellierungsmöglichkeiten von Zeitreihen</li> <li>• Erwerb von Methoden aus Zeitreihen symbolische Musterbeschreibungen erzeugen zu können</li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>• praktische Verwendung von explorativen statistischen Methoden zur Beschreibung und Analyse der Daten</li> <li>• Theorie und Praxis von Fouriertransformationen für Zeitreihen</li> <li>• Theorie und Praxis von Wavelettransformationen für Zeitreihen</li> <li>• Modellierung Stochastischer Prozesse (ARMA, GARCH)</li> <li>• Markov Modelle</li> <li>• Neuronale Netze zur Analyse und Prognose von Zeitreihen</li> <li>• Temporales Knowledge Discovery</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 3 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben und aktive Teilnahme an den Übungen; Bestehen von Zwischentests gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Vorlesungen Praktische Informatik I und II, Knowledge Discovery (als Empfehlung)
<b>Arbeitsaufwand</b>	Präsenzzeit 75 Std., Selbststudium 135 Std.
<b>Leistungspunkte</b>	7
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur. Note: Note der Abschlussklausur
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, jedes Sommersemester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 692 – Künstliche Intelligenz II</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Kenntnis der aktuellen KI-Methoden</li> <li>• nichtmonotone Logiken</li> <li>• Agentensysteme</li> <li>• Einführung in die Robotik</li> <li>• Distributive und konnektionistische Methoden</li> <li>• Temporal Reasoning</li> </ul>
<b>Thema und Inhalt</b>	<ol style="list-style-type: none"> <li>1. Agentensysteme</li> <li>2. Wahrnehmen und Agieren in technischen Systemen</li> <li>3. autonome Systeme (Robotik)</li> <li>4. Künstliches Leben</li> <li>5. Evolutionäre Algorithmen</li> <li>6. Temporale Kalküle</li> <li>7. Nichtklassische Logiken</li> </ol>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 2 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben und aktive Teilnahme an den Übungen gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Vorlesungen Praktische Informatik I und II
<b>Arbeitsaufwand</b>	Präsenzzeit 60 Std., Selbststudium 120 Std
<b>Leistungspunkte</b>	6
<b>Art der Prüfungen</b>	Erfolgreiches Absolvieren eines Kolloquiums oder einer Klausur Note: Note des Kolloquiums/der Klausur
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, jedes zweite Sommersemester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

- **Bereich Theoretische Informatik**

<b>Modulbezeichnung</b>	<b>CS 509 – Formale Methoden und Grundlagen der Programmierung</b>
<b>Kompetenzen und Qualifikationsziel</b>	Ziel des Moduls ist die Vermittlung neuer Forschungsergebnisse aus aktuellen wissenschaftlichen Beiträgen in nationalen und internationalen Fachzeitschriften sowie von Konferenzen.
<b>Thema und Inhalt</b>	Es werden Inhalte der neueren Forschung aus dem Bereich Formale Methoden und Grundlagen der Programmierung behandelt. Die spezielle fachliche Ausrichtung des Moduls wird vom Dozenten festgelegt.
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 3 SWS und Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Kenntnisse aus den Vorlesungen Praktische Informatik I – II
<b>Arbeitsaufwand</b>	Präsenzzeit 75 Std , Selbststudium 135 Std
<b>Leistungspunkte</b>	7
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder eines Kolloquiums Note: Die Note entspricht der Klausur- bzw. Kolloquiumsnote.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, unregelmäßig
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 521 – Model Checking</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Spezifikation von Hardware und Protokollen</li> <li>• Umgang mit Temporaler Logiken</li> <li>• Umgang mit Model Checkern</li> <li>• Verständnis zugrundeliegender Algorithmen</li> </ul>
<b>Thema und Inhalt</b>	<p>Model Checking ist eine Technik der automatischen Verifikation von Hard- und Softwaresystemen. Sie wird industriell zur Verifikation von Bauteilen und von Protokollen eingesetzt.</p> <ul style="list-style-type: none"> <li>• Modellierung von Hardware und von Protokollen durch Kripke Strukturen,</li> <li>• Zeitabhängige Eigenschaften,</li> <li>• Safety, Liveness, Fairness.</li> <li>• Computation Tree Logic (CTL)</li> <li>• Umgang mit SMV</li> <li>• Model Checking Algorithmen für CTL</li> <li>• Lineare Temporale Logik (LTL)</li> <li>• Büchi-Automaten, Algorithmen für LTL ModelChecking</li> <li>• effiziente Darstellung von Schaltkreisen: OBDDs</li> <li>• Symbolisches Model Checking</li> <li>• Kompositionalität: Assume-Commitment Beweise</li> <li>• Bounded Model Checking</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Modulen Theoretische Informatik und Logik und Diskrete Mathematik
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std., Selbststudium 180 Std.
<b>Leistungspunkte</b>	9
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder Kolloquiums Note: Note der Abschlussprüfung (Klausur oder Kolloquium)
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle drei bis vier Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge



<b>Modulbezeichnung</b>	<b>CS 522 – Rechnergestützte Beweissysteme</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Spezifikation formaler Beweisaufgaben</li> <li>• Methoden, Kalküle und Algorithmen rechnergestützten Beweises</li> <li>• Umgang mit praktischen Beweissystemen</li> <li>• Spezielle Logiken</li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>• Gentzenkalkül für Aussagenlogik</li> <li>• Prädikatenlogik, Behandlung von Quantoren, Resolventenmethode,</li> <li>• Spezifizieren und Beweisen in PVS</li> <li>• Typsysteme</li> <li>• Gleichheit, Termersetzungssysteme,</li> <li>• Entscheidungsprozeduren, Nelson-Oppen, Shostak-Algorithmus</li> <li>• Induktion, Logik höherer Stufe</li> <li>• Synthese von Programmen und Datentypen</li> <li>• Co-Datentypen</li> <li>• Intuitionistische Logik,</li> <li>• Implementierung von Nichtstandard Logiken in Jape</li> <li>• Hardwarekonstruktion als Beweis : Das Lambda-System</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Modulen Theoretische Informatik sowie Logik und Diskrete Mathematik
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std., Selbststudium 180 Std.
<b>Leistungspunkte</b>	9
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder Kolloquiums Note: Note der Abschlussprüfung
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle drei bis vier Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 523 – Berechenbarkeit und Beweisbarkeit</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Vertiefung der Berechenbarkeitstheorie</li> <li>• Illustration und Anwendung in <ul style="list-style-type: none"> <li>○ Programmiersprachen</li> <li>○ Logik</li> <li>○ Algebra</li> </ul> </li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>• Berechenbarkeitskonzepte</li> <li>• Definierbarkeit, Beweisbarkeit</li> <li>• Unmöglichkeitsbeweise</li> <li>• Gödelscher Unvollständigkeitssatz</li> <li>• Lambda-Kalkül, Kombinatorische Logik</li> <li>• Objektkalkül (Featherweight Java)</li> <li>• Intuitionistische Logik</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Modulen Praktische Informatik I – III, Theoretische Informatik sowie Logik und Diskrete Mathematik
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std., Selbststudium 180 Std.
<b>Leistungspunkte</b>	9
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder Kolloquiums Note: Note der Abschlussprüfung
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, unregelmäßig
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 552 – Semantik von Programmiersprachen</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Erlernen und Anwenden von verschiedenen Techniken der Semantikgebung</li> <li>• Anwendung formaler Methoden zur Beschreibung und Analyse von programmiersprachlichen Konstrukten</li> </ul>
<b>Thema und Inhalt</b>	<p>Es werden verschiedene Ansätze zur formalen Beschreibung der Bedeutung (Semantik) von Programmiersprachen vorgestellt. Semantikmodelle unterstützen das Verständnis von Programmiersprachen und die Entwicklung von Compilern. Grundsätzlich unterscheidet man drei verschiedene Ansätze, die in der Vorlesung anhand einer einfachen imperativen Modellsprache eingeführt werden:</p> <ul style="list-style-type: none"> <li>◆ Operationelle Semantiken: Beschreibung von Berechnungen</li> <li>◆ Denotationelle Semantiken: Beschreibung des Ein-/Ausgabeverhaltens</li> <li>◆ Axiomatische Semantiken: Angabe von Aussagen über Programmeigenschaften</li> </ul> <p>Es werden auch weiterführende Techniken für funktionale und objektorientierte Sprachen und die Beschreibung von Nichtdeterminismus und Parallelität behandelt.</p>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 3 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Vorlesungen Praktische Informatik I-III
<b>Arbeitsaufwand</b>	Präsenzzeit 75 Std., Selbststudium 165 Std.
<b>Leistungspunkte</b>	8
<b>Art der Prüfungen</b>	Bestehen einer Abschlussprüfung (Klausur oder Kolloquium) Note: Note der Abschlussprüfung
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle drei bis vier Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 609 – Formale Methoden und Grundlagen der Programmierung II</b>
<b>Kompetenzen und Qualifikationsziel</b>	Ziel des Moduls ist die Vermittlung fortgeschrittener neuer Forschungsergebnisse aus aktuellen wissenschaftlichen Beiträgen in nationalen und internationalen Fachzeitschriften sowie von Konferenzen.
<b>Thema und Inhalt</b>	Es werden Inhalte der neueren Forschung aus dem Bereich Formale Methoden und Grundlagen der Programmierung behandelt. Die spezielle fachliche Ausrichtung des Moduls wird vom Dozenten festgelegt.
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 3 SWS und Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Kenntnisse aus den Vorlesungen Praktische Informatik I – II
<b>Arbeitsaufwand</b>	Präsenzzeit 75 Std , Selbststudium 135 Std
<b>Leistungspunkte</b>	7
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder eines Kolloquiums Note: Die Note entspricht der Klausur- bzw. Kolloquiumsnote
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, unregelmäßig
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 621 – Abstrakte Datentypen</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Beschreibung von Datenstrukturen</li> <li>• Abstrakte Datentypen, Morphismen, abgeleitete Strukturen</li> <li>• Freiheit, Initialität und Induktion</li> <li>• Gleichungsspezifikationen</li> <li>• Mehrsortige Systeme</li> <li>• Hidden Specifications</li> </ul>
<b>Thema und Inhalt</b>	<p>Universelle Algebra ist die mathematische Theorie abstrakter Datentypen. Diese Vorlesung bereitet den theoretischen Hintergrund für Entwurf, Spezifikation und Verifikation von Datentypen.</p> <ul style="list-style-type: none"> <li>• Typen, Algebren, Morphismen</li> <li>• Unterstrukturen, Kongruenzen, Produkte, Bilder</li> <li>• Terme und Gleichungen</li> <li>• Initiale und freie Objekte</li> <li>• Der Satz von Birkhoff</li> <li>• Maltsev Bedingungen</li> <li>• Mehrsortige Algebren</li> <li>• Hidden sorts, hidden Specifications</li> <li>• Verhaltensspezifikationen</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Modulen Theoretische Informatik sowie Logik und Diskrete Mathematik
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std., Selbststudium 180 Std.
<b>Leistungspunkte</b>	9
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder Kolloquiums Note: Note der Abschlussprüfung
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle drei bis vier Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

<b>Modulbezeichnung</b>	<b>CS 622 – Zustandsbasierte Systeme</b>
<b>Kompetenzen und Qualifikationsziel</b>	<p>Coalgebren sind mathematische Strukturen, mit denen die Semantik verschiedenster formaler Systeme der Informatik beschrieben werden kann. Dazu gehören u.a.</p> <ul style="list-style-type: none"> <li>• Transitionssysteme (Bisimulation, Beobachtungsäquivalenz),</li> <li>• Automaten (deterministisch, nichtdeterministisch, mit und ohne Ausgabe),</li> <li>• Objektorientierte Programmiersprachen (finale Datentypen, unendliche Objekte, Ströme).</li> </ul> <p>Die mathematische Theorie der Coalgebren wird entwickelt, wobei zahlreiche Anwendungsbeispiele aus der Informatik herangezogen werden.</p>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>• Beschreibung zustandsbasierter Systeme</li> <li>• Kategorientheoretische Abstraktionen</li> <li>• Strukturtheorie</li> <li>• Bisimulationen und Verhaltensäquivalenz</li> <li>• Corekursive Definitionen, Coinduktive Verifikation</li> <li>• Terminale und Cofreie Systeme.</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 4 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Modulen Theoretische Informatik sowie Logik und Diskrete Mathematik
<b>Arbeitsaufwand</b>	Präsenzzeit 90 Std., Selbststudium 180 Std.
<b>Leistungspunkte</b>	9
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder Kolloquiums Note: Note der Abschlussprüfung
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle drei bis vier Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramts- und im Bachelorstudiengang Informatik

<b>Modulbezeichnung</b>	<b>CS 651 - Strukturen funktionaler Programmiersprachen</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Vertiefung des Verständnisses funktionaler Programmierung durch Erarbeitung der zugrunde liegenden formalen Modelle</li> <li>• Erarbeitung von Erweiterungen funktionaler Sprachen</li> <li>• Erlernen von Methoden zur Implementierung von mächtigen Sprachkonstrukten mit abstrakten Maschinen</li> </ul>
<b>Thema und Inhalt</b>	Lambda-Kalkül, Typsysteme, denotationelle und operationelle Semantik, Implementierungstechniken, Kombinatorbibliotheken, Monaden, Generische Programmierung, Meta-Programmierung, Exemplarische Anwendungen
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 3 SWS, Übungen 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus der Vorlesung Praktische Informatik III
<b>Arbeitsaufwand</b>	Präsenzzeit 75 Std., Selbststudium 165 Std.
<b>Leistungspunkte</b>	8
<b>Art der Prüfungen</b>	Bestehen einer Abschlussprüfung (Klausur oder Kolloquium) Note: Note der Abschlussprüfung
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle drei bis vier Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge

- Seminar

<b>Modulbezeichnung</b>	<b>CS 600 – Vertiefung / spezielle Themen der Informatik (Seminar)</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Erwerb von Fähigkeiten und Techniken des wissenschaftlichen Arbeitens;</li> <li>• Erwerb von Fähigkeiten und Fertigkeiten beim Lesen und Reproduzieren/Zusammenfassen wissenschaftlicher Texte;</li> <li>• Ausbau von Fähigkeiten der mündlichen Präsentation wissenschaftlicher Abhandlungen in Form eines Vortrags und durch die aktive Teilnahme an der Diskussion zu anderen Vorträgen</li> </ul>
<b>Thema und Inhalt</b>	Themen aus dem Bereich der Informatik. Die spezielle fachliche Ausrichtung des Seminars wird vom betreuenden Dozenten festgelegt.
<b>Organisations-, Lehr- und Lernform</b>	Seminar 2 SWS Erfolgreiche Ausarbeitung eines Themas, Vortragspräsentation gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden abhängig von der fachlichen Ausrichtung des Seminars, generell jedoch Kenntnisse aus den Modulen Praktische Informatik I-III und Mathematik I/Lineare Algebra
<b>Arbeitsaufwand</b>	Präsenzzeit 30 Std., Selbststudium 60 Std.
<b>Leistungspunkte</b>	3
<b>Art der Prüfungen</b>	schriftliche Ausarbeitung Note: Die Note ergibt sich aus der Note der Modulprüfung.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, in jedem Semester
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik; Exportmodul für andere Studiengänge



## 11.3.5 Fachdidaktikmodule

<b>Modulbezeichnung</b>	<b>CS 601 – Didaktik der Informatik (Vertiefung)</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• didaktische Aufbereitung fachwissenschaftlicher Inhalte</li> <li>• Einsatz spezifischer Lehr-/Lernformen für die Wissensvermittlung</li> <li>• Einschätzung und Vermittlung der Stellung des Unterrichtsgegenstandes in der Fachwissenschaft, im interdisziplinären Zusammenhang und in der Alltagswelt</li> </ul>
<b>Thema und Inhalt</b>	<p>Mögliche Themen sind:</p> <ul style="list-style-type: none"> <li>• Dynamische Datenstrukturen in der Schule</li> <li>• Objektorientiertes Modellieren</li> <li>• Informatikunterricht in der Sekundarstufe I / im Wahlpflichtbereich</li> <li>• Prolog in der Schule</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung/Seminar 2 SWS Erfolgreiche Ausarbeitung eines Themas, Vortragspräsentation und schriftliche Ausarbeitung im Seminar gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse abhängig von der fachlichen Ausrichtung der Veranstaltung
<b>Arbeitsaufwand</b>	Präsenzzeit 30 Std., Selbststudium 60 Std.
<b>Leistungspunkte</b>	3
<b>Art der Prüfungen</b>	Abschlussprüfung Note: Die Note ergibt sich aus der Note der Modulprüfung
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, jedes zweite Sommersemester
<b>Verwendbarkeit des Moduls</b>	Pflichtmodul im Lehramtsstudiengang Informatik

<b>Modulbezeichnung</b>	<b>CS 602 – Schulpraktische Studien II im Fach Informatik</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>○ Die Studierenden sollen fachdidaktische Ziele und Inhalte in ihrem konkreten Bezug zum Schulalltag und <i>im</i> Schulalltag kennen lernen.</li> <li>○ Die Hospitationsphase soll es ermöglichen, die verschiedenen Unterrichts-komponenten (Motivation, Unterrichtsphasen, Leistungsmessung, ...) an „konkreten Schülerpersönlichkeiten“ zu erkennen und zu diskutieren.</li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>○ Informatik als Unterrichtsfach an allgemeinbildenden Schulen</li> <li>○ Vom Lehrplan zur Planung einer Unterrichtseinheit</li> <li>○ Entwurf einer Unterrichtssequenz</li> <li>○ Planung einer Unterrichtsstunde</li> <li>○ Phasen einer Unterrichtsstunde, Feinplanung</li> <li>○ Der Stundenentwurf</li> <li>○ Unterrichtsbeobachtung und Unterrichtsbesprechung/ -reflexion</li> <li>○ Leistungsmessung und -bewertung</li> <li>○ Entwurf und Korrektur einer Klassenarbeit</li> <li>○ „mündliche“ Noten</li> <li>○ Unterricht im Vergleich</li> <li>○ Die Arbeit im Computerlabor</li> <li>○ Informatik-Projekte</li> <li>○ Medieneinsatz</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Seminar 2 SWS und Schulpraktikum (Hospitation) 50 Unterrichtsstunden, Hausarbeit
<b>Teilnahmevoraussetzungen</b>	Schulpraktische Studien I Erwartet werden Grundkenntnisse aus den Vorlesungen Praktische Informatik I und II, Datenbanksysteme, Einführung in die Didaktik der Informatik,
<b>Arbeitsaufwand</b>	Präsenzzeit 60 Std., Selbststudium 90 Std.
<b>Leistungspunkte</b>	5
<b>Art der Prüfungen</b>	Erarbeitung einer Unterrichtseinheit, mindestens drei eigene Unterrichtsversuche, Hausarbeit (Praktikumsbericht mit einer eingereichten Stunde) Note: Die Note ergibt sich aus der Note der Modulprüfung.
<b>Dauer des Moduls und Angebotsturnus</b>	1-2 Semester; das Modul wird jedes zweite Semester angeboten.
<b>Verwendbarkeit des Moduls</b>	Wahlpflichtmodul im Lehramtsstudiengang Informatik

<b>Modulbezeichnung</b>	<b>CS 603 – Kompensationsmodul zu Schulpraktischen Studien II im Fach Informatik</b>
-------------------------	--

<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>○ Die Studierenden sollen fachdidaktische Ziele und Inhalte in ihrem konkreten Bezug zum Schulalltag und <i>im</i> Schulalltag kennen lernen.</li> <li>○ Die Hospitationsphase soll es ermöglichen, die verschiedenen Unterrichts-komponenten (Motivation, Unterrichtsphasen, Leistungsmessung, ...) an „konkreten Schülerpersönlichkeiten“ zu erkennen und zu diskutieren.</li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>○ Informatik als Unterrichtsfach an allgemeinbildenden Schulen</li> <li>○ Vom Lehrplan zur Planung einer Unterrichtseinheit</li> <li>○ Entwurf einer Unterrichtssequenz</li> <li>○ Planung einer Unterrichtsstunde</li> <li>○ Phasen einer Unterrichtsstunde, Feinplanung</li> <li>○ Der Stundenentwurf</li> <li>○ Unterrichtsbeobachtung und Unterrichtsbesprechung/ -reflexion</li> <li>○ Leistungsmessung und -bewertung</li> <li>○ Entwurf und Korrektur einer Klassenarbeit</li> <li>○ „mündliche“ Noten</li> <li>○ Unterricht im Vergleich</li> <li>○ Die Arbeit im Computerlabor</li> <li>○ Informatik-Projekte</li> <li>○ Medieneinsatz</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	fachdidaktisches Seminar mit Praxis berücksichtigendem Unterrichtsbezug
<b>Teilnahmevoraussetzungen</b>	Schulpraktische Studien I Erwartet werden Grundkenntnisse aus den Vorlesungen Praktische Informatik I und II, Datenbanksysteme, Einführung in die Didaktik der Informatik,
<b>Arbeitsaufwand</b>	Präsenzzeit 50 Std., Selbststudium 100 Std.
<b>Leistungspunkte</b>	5
<b>Art der Prüfungen</b>	Seminarvortrag (Gestaltung einer Seminarsitzung mit schriftlicher Ausarbeitung), schriftlicher Bericht über die zusätzliche unterrichtsbezogene Projektarbeit Note: Die Note ergibt sich aus der Note der Modulprüfung.
<b>Dauer des Moduls und Angebotsturnus</b>	1-2 Semester; das Modul wird jedes zweite Semester angeboten.
<b>Verwendbarkeit des Moduls</b>	Pflichtmodul im Lehramtsstudiengang Informatik

<b>Modulbezeichnung</b>	<b>CS 604 – Proseminar in Informatik</b>
<b>Kompetenzen und Qualifikationsziel</b>	<p>Studierende sollen in das selbständige wissenschaftliche Arbeiten eingeführt werden und dabei</p> <ul style="list-style-type: none"> <li>○ an Hand von Literaturvorgaben in einer kleinen Gruppe Vorschläge für die Stoffaufteilung und Ergänzungen machen</li> <li>○ den eigenen, ca. einstündigen Vortrag vorbereiten und weitgehend frei und für die Seminarteilnehmer gut nachvollziehbar abhalten</li> <li>○ eine Ausarbeitung zum Vortrag anfertigen und</li> <li>○ sich aktiv an der Diskussion der Vorträge beteiligen.</li> </ul> <p>Die Studierenden sollen dabei</p> <ul style="list-style-type: none"> <li>○ ihre mündliche Kommunikationsfähigkeit durch das Einüben der freien Rede vor einem größeren Publikum und der Diskussion mit diesem verbessern</li> <li>○ während der Vorbereitung Erfahrungen in Teamarbeit und Arbeitsorganisation (Stoffauswahl, Hilfsmittel, Zeiteinteilung) sammeln</li> <li>○ geeignete Präsentationsmittel auswählen und den Einsatz von Präsentationstechniken erlernen.</li> </ul>
<b>Thema und Inhalt</b>	Themen aus dem Bereich der Pflichtmodule der Informatik. Die spezielle fachliche Ausrichtung des Seminars wird vom betreuenden Dozenten festgelegt.
<b>Organisations-, Lehr- und Lernform</b>	Proseminar 3 SWS Schriftliche Ausarbeitung und Vortragspräsentation gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden abhängig von der fachlichen Ausrichtung des Seminars, generell jedoch Grundkenntnisse aus den Vorlesungen Praktische Informatik I-III und Mathematik I/Lineare Algebra
<b>Arbeitsaufwand</b>	Präsenzzeit 40 Std., Selbststudium 80 Std.
<b>Leistungspunkte</b>	4
<b>Art der Prüfungen</b>	Schriftliche Ausarbeitung Note: Die Note ergibt sich aus der Modulprüfung.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, in jedem Semester
<b>Verwendbarkeit des Moduls</b>	Fachdidaktisches Wahlpflichtmodul im Lehramtsstudiengang Informatik

<b>Modulbezeichnung</b>	<b>CS 605 – Tutorenseminar in Informatik</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• didaktische Aufbereitung fachwissenschaftlicher Inhalte</li> <li>• Einsatz spezifischer Lehr-/Lernformen für die Wissensvermittlung</li> <li>• Einschätzung und Vermittlung der Stellung des Unterrichtsgegenstandes in der Fachwissenschaft</li> </ul>
<b>Thema und Inhalt</b>	Die Studierenden sollen in einem Seminar spezifische Lehr-/Lernformen diskutieren und den Einsatz eines Stoffgebietes als Unterrichtsgegenstand untersuchen. Sie können in den begleitenden fachdidaktischen Übungen der Vorlesungen Praktische Informatik II, Theoretische Informatik oder Datenbanksysteme als Tutoren eingesetzt werden und auf diese Weise die didaktische Aufbereitung des jeweiligen Stoffes erproben.
<b>Organisations-, Lehr- und Lernform</b>	Seminar 1 SWS, Tutorium 2 SWS Schriftliche Ausarbeitung und Vortragspräsentation gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus der Einführung in die Didaktik der Informatik und der Didaktik der Informatik – Vertiefung; Modul, in dem der Einsatz als Tutor erfolgt
<b>Arbeitsaufwand</b>	Präsenzzeit 40 Std., Selbststudium 80 Std.
<b>Leistungspunkte</b>	4
<b>Art der Prüfungen</b>	schriftliche Ausarbeitung Note: Die Note ergibt sich aus der Note der Modulprüfung.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig im Sommersemester
<b>Verwendbarkeit des Moduls</b>	Fachdidaktisches Wahlpflichtmodul im Lehramtsstudiengang

<b>Modulbezeichnung</b>	<b>CS 502 - Information, Informationssysteme, Informationsgesellschaft</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Grundverständnis für die Bedeutung Verankerung und interdisziplinäre Verwendung des Informationsbegriffs</li> <li>• Kenntnisse über die Grundlagen, die Entwicklung und den Einsatz von Informationssystemen</li> <li>• Verständnis und kritische Reflexion des Schlagworts "Informationsgesellschaft", seiner gesellschaftlichen Rolle, Einbettung und wesentlichen Bezüge</li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>• Informationsbegriff: Semiotische Fundierung, Information als interdisziplinäres Phänomen</li> <li>• Informationssysteme: Systembegriff, Klassifizierung von Systemen, Eigenschaften und Modelle für die Entwicklung von Informationssystemen, Informationssysteme im gesellschaftlichen Umfeld</li> <li>• Informationsgesellschaft: Informationstechnik in den Bereichen Heim, Arbeit, Wirtschaft/Industrie, Staat/Kommune; Informatik-Berufsfelder, Risiken der Informationsgesellschaft, Informations-Überflutung, Verantwortung des Informatikers, Zukunftsvisionen</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 2 SWS, Seminar 2 SWS Erfolgreiche Teilnahme am Begleitseminar, selbständige Bearbeitung eines Seminarthemas, Halten eines Seminarvortrags gem. Modulankündigung
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Vorlesungen Praktische Informatik I–III
<b>Arbeitsaufwand</b>	Präsenzzeit 40 Std., Selbststudium 80 Std.
<b>Leistungspunkte</b>	4
<b>Art der Prüfungen</b>	Abfassen einer Seminararbeit. Note: Die Note ergibt sich aus der Note der Modulprüfung.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle 3 bis 4 Semester
<b>Verwendbarkeit des Moduls</b>	Fachdidaktisches Wahlpflichtmodul im Lehramtsstudiengang

<b>Modulbezeichnung</b>	<b>CS 503 – Neuere Entwicklungen der Fachdidaktik Informatik</b>
<b>Kompetenzen und Qualifikationsziel</b>	Ziel des Moduls ist die Vermittlung neuer Forschungsergebnisse aus aktuellen wissenschaftlichen Beiträgen in nationalen und internationalen Fachzeitschriften sowie von Konferenzen.
<b>Thema und Inhalt</b>	Es werden Inhalte der neueren Forschung aus dem Bereich der Fachdidaktik Informatik behandelt. Die spezielle fachliche Ausrichtung des Moduls wird vom Dozenten festgelegt.
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 2 SWS und Übungen oder Seminar 2 SWS Erfolgreiche Bearbeitung von Übungsaufgaben bzw. erfolgreiche Teilnahme am Begleitseminar (Vortrag und Ausarbeitung)
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Kenntnisse aus den Vorlesungen Praktische Informatik I – II
<b>Arbeitsaufwand</b>	Präsenzzeit 40 Std., Selbststudium 80 Std.
<b>Leistungspunkte</b>	4
<b>Art der Prüfungen</b>	Bestehen einer Abschlussklausur oder eines Kolloquiums Note: Die Note entspricht der Klausur- bzw. Kolloquiumsnote
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, unregelmäßig
<b>Verwendbarkeit des Moduls</b>	Fachdidaktik Wahlpflichtmodul im Lehramtsstudiengang Informatik

<b>Modulbezeichnung</b>	<b>CS 606 - Gesellschaftliche Aspekte der Informatik</b>
<b>Kompetenzen und Qualifikationsziel</b>	<ul style="list-style-type: none"> <li>• Verständnis und kritische Reflexion der Informatik und ihrer gesellschaftlichen Rolle</li> </ul>
<b>Thema und Inhalt</b>	<ul style="list-style-type: none"> <li>• Informatik in den Bereichen Heim, Arbeit, Wirtschaft/Industrie, Staat/Kommune; Informatik-Berufsfelder, Risiken der Informationsgesellschaft, Informations-Überflutung, Verantwortung des Informatikers, Zukunftsvisionen</li> </ul>
<b>Organisations-, Lehr- und Lernform</b>	Vorlesung 2 SWS, Übungen und/oder Seminar 2 SWS Erfolgreiche Teilnahme an den Übungen bzw. am Begleitseminar, selbständige Bearbeitung eines Seminarthemas,
<b>Teilnahmevoraussetzungen</b>	Erwartet werden Grundkenntnisse aus den Vorlesungen Praktische Informatik I–III
<b>Arbeitsaufwand</b>	Präsenzzeit 40 Std., Selbststudium 80 Std.
<b>Leistungspunkte</b>	4
<b>Art der Prüfungen</b>	Halten eines Seminarvortrags und Abfassen einer Seminararbeit oder Bestehen einer Klausur oder eines Kolloquiums Note: Die Note ergibt sich aus der Note der Modulprüfung.
<b>Dauer des Moduls und Angebotsturnus</b>	1 Semester, regelmäßig alle 3 bis 4 Semester
<b>Verwendbarkeit des Moduls</b>	Fachdidaktisches Wahlpflichtmodul im Lehramtsstudiengang