

Modulbeschreibungen Q-Master ISS/Gym

Informatik

Im Bereich Fachdidaktik sind folgende Module zu absolvieren:

Modul: Grundlagen und Vertiefung Fachdidaktik Informatik im Profil Quereinstieg 3

Modul: Vertiefung Fachdidaktik Informatik im Profil Quereinstieg 5

Student*innen des Studienfachs Informatik als Fach 1 absolvieren zudem:

Modul: Schulpraktische Studien im Unterrichtsfach Informatik Fach 1 6

Student*innen des Studienfachs Informatik als Fach 2 absolvieren zudem:

Modul: Schulpraktische Studien im Unterrichtsfach Informatik Fach 2 8

Student*innen des Studienfachs Informatik als Fach 2 absolvieren außerdem

Module im Umfang von 10 LP aus dem folgenden Angebot:

Modul: Bildverarbeitung 10

Modul: Computergrafik 11

Modul: Computer-Vision 12

Modul: Datenbanktechnologie 13

Modul: Grundlagen des Softwaretestens 14

Modul: Künstliche Intelligenz 15

Modul: Medizinische Bildverarbeitung 16

Modul: Modellgetriebene Softwareentwicklung 17

Modul: Netzbasierte Informationssysteme 18

Modul: Rechnersicherheit 19

Modul: Übersetzerbau 20

Modul: XML - Technologien 21

Modul: Praktiken professioneller Softwareentwicklung 22

Modul: Semantik von Programmiersprachen 23

Modul: Betriebssysteme 24

Modul: Robotik 25

Aus dem folgenden Angebot sind Module im Umfang von 25 LP zu absolvieren:

Modul: Objektorientierte Programmierung für Studentinnen und Studenten mit Programmierkenntnissen 26

Modul: Objektorientierte Programmierung für Studentinnen und Studenten ohne Programmierkenntnissen 27

Modul: Datenbanksysteme 28

Modul: Grundlagen der theoretischen Informatik 29

Modul: Softwaretechnik 30

Modul: Auswirkungen der Informatik 31

Modul: Gesellschaftliche Aspekte der Informatik 32

Modul: Softwareprojekt A	33
Modul: Softwareprojekt B.....	34
Modul: Rechnerarchitektur.....	35
Modul: Betriebs- und Kommunikationssysteme.....	36
Modul: Grundlagen der technischen Informatik.....	37
Modul: Algorithmen, Datenstrukturen und Datenabstraktion A.....	38

Vertiefende Information, u.a. zum Studienverlauf, finden Sie in den entsprechenden **Studien- und Prüfungsordnungen**. Die jeweilige SPO ist für jedes Modul als Fußnote ausgewiesen.



Modul: Grundlagen und Vertiefung Fachdidaktik Informatik im Profil Quereinstieg¹

Modul: Grundlagen und Vertiefung Fachdidaktik Informatik im Profil Quereinstieg
Hochschule/Fachbereich/Lehreinheit: Freie Universität Berlin/Mathematik und Informatik/Informatik
Modulverantwortliche/r: Professur für Informatikdidaktik
Zugangsvoraussetzungen: Keine
Qualifikationsziele: Die Studentinnen und Studenten sind in der Lage, <ul style="list-style-type: none">– die eigene Lernbiografie im Fach Informatik sowie die eigenen fachlichen Lernprozesse systematisch und theoriegeleitet zu reflektieren und die mitgebrachten subjektiven Theorien zu überprüfen und zu erweitern,– informatikdidaktische Konzeptionen strukturiert und systematisch darzustellen und zu erläutern,– ausgewählte informatikdidaktische Theorien und Konzeptionen exemplarisch auf fachunterrichtliche und außerschulische Lernorte zu beziehen und zu beurteilen,– informatikdidaktische Theorien und Konzeptionen in einen Zusammenhang mit fachlichem Lehren und Lernen zu stellen,– fachwissenschaftliche und bildungswissenschaftliche Erkenntnisse unter informatikdidaktischer Perspektive auf ausgewählte außerschulische Lernorte zu beziehen,– mit gender- und diversityorientierten sowie inklusionspädagogischen Prinzipien bezogen auf den Informatikunterricht umzugehen,– Kriterien der fachlichen und fachübergreifenden Kommunikation darzulegen,– exemplarisch informatische und fachübergreifende Themen mit Studentinnen und Studenten, Fachpersonen und anderen fachlich Interessierten ziel- und adressatengerecht kommunizieren,– Informatikunterricht und Curricula unter informatikdidaktischer Perspektive zu analysieren,– Schulbücher, Lehr- und Lernmaterialien für das Fach Informatik unter Diversity- und Inklusionsaspekten zu analysieren und zu beurteilen.
Inhalte: Das Studium befasst sich mit: <ul style="list-style-type: none">– Theoretische Perspektiven: Dazu gehören Bildungswert des Faches, Bildungsstandards/Curricula, Inhalte und besondere Methoden des Informatikunterrichts, Wege der Erkenntnisgewinnung im Fach Informatik, Lerntheorien und Schülervorstellungen, informatikdidaktische Prinzipien und Unterrichtsmethoden, Curriculumentwicklung, Geschichte des Faches, Kompetenzorientierung, Heterogenität, Diversity und Inklusion, Leistungsbeurteilung, Analyse von Lehr- und Lernmedien– Praktische Perspektiven: Dazu gehören Berufsfeldorientierung, Gestaltung von Lernumgebungen unter besonderer Berücksichtigung praktischer Elemente des Lernens im Fach Informatik, Experimente im Unterricht; Umgang mit Modellen, Einsatz neuer Medien für Lehren und Lernen, Schulversuche, reflektierte Anwendung von Medien und Methoden zur Vermittlung von Informatik (Rollenspiele; Gruppenarbeit ...).– Erste fachspezifische Praxisbegegnungen: Dazu gehören angeleitete Planung von experimentell ausgerichteten Unterrichtssequenzen in Partner- und Gruppenarbeit; unterrichtspraktische Übungen, Micro- und Peer-Teaching, Teaching-Experiments; gemeinsame Reflexion des Lehrens und Lernens fachbezogener Inhalte und Konzepte.

¹ Studien- und Prüfungsordnung der Freien Universität Berlin für den Masterstudiengang für das Lehramt an Integrierten Sekundarschulen und Gymnasien mit dem Profil Quereinstieg



Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Seminar I	2	Bearbeitung von Präsenzaufgaben, aktive Beteiligung, an Diskussionen, Teaching-Experiments	Präsenzzeit S I 30
Seminar II	2		Vor- und Nachbereitung S 30
			Präsenzzeit S II 30
			Vor- und Nachbereitung S II 30
			Prüfungsvorbereitung und Prüfung 30
Modulprüfung:		Klausur (60 Minuten) oder mündliche Prüfung (ca. 20 Minuten) oder schriftliche Ausarbeitung (ca. 10 Seiten)	
Modulsprache:		Deutsch	
Pflicht zur regelmäßigen Teilnahme:		Ja	
Arbeitsaufwand insgesamt:		150 Stunden	5 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Einmal im Jahr	
Verwendbarkeit:		Masterstudiengang für das Lehramt an Integrierten Sekundarschulen und Gymnasien mit dem Profil Quereinstieg	



Modul: Vertiefung Fachdidaktik Informatik im Profil Quereinstieg²

Modul: Vertiefung Fachdidaktik Informatik im Profil Quereinstieg									
Hochschule/Fachbereich/Lehreinheit: Freie Universität Berlin/Mathematik und Informatik/Informatik									
Modulverantwortliche/r: Professur für Informatikdidaktik									
Zugangsvoraussetzungen: Keine									
Qualifikationsziele:									
Die Studentinnen und Studenten verfügen über die folgenden schulartbezogenen Kompetenzen, sie können:									
<ul style="list-style-type: none"> – Informatikdidaktische Theorien und Konzeptionen rezipieren, reflektieren und auf schulische, fachunterrichtliche und außerschulische Lernorte und Felder beziehen, – schulformbezogen Informatikdidaktische Konzeptionen strukturiert und systematisch darstellen und erläutern sowie theoriegeleitet in einen schulformspezifischen Zusammenhang mit fachlichem Lehren und Lernen stellen, – Informatikdidaktische Theorien und Konzeptionen auf fach- und bildungswissenschaftliche Ansätze beziehen und dabei die Aspekte Gender, Diversity und Inklusion auf Lerngruppen bezogen berücksichtigen und bewerten, – auf Basis der Bildungsstandards für das Fach Informatik schulformbezogen exemplarisch Ansätze selbstbestimmten, kooperativen, kumulativen sowie kontextbezogenen Lernens erörtern, – den Einfluss des fachspezifischen Medieneinsatzes auf das Lehren und Lernen von Informatik reflektieren und bewerten. 									
Inhalte:									
Es werden unterschiedliche Schwerpunkte angeboten, wie beispielsweise: spezielle Probleme des schulartbezogenen Lehrens und Lernens von Informatik, spezielle experimentelle Lernumgebungen und experimentelle Zugänge zu ausgewählten Themen; Bildung für Nachhaltige Entwicklung (BNE), Gestaltung und Analyse von lernförderlichen Aufgaben; Differenzierung und Umgang mit Heterogenität, Informatiklernen an außerschulischen Lernorten und im Schülerlabor, fachübergreifender Unterricht, unterrichtspraktische Erfahrungen in komplexitätsreduzierten Lehr-Lernsituationen im Lehr-Lern-Labor/Schülerlabor, Gender und Diversity im Informatikunterricht.									
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)						
Seminar	3	Aktive Beteiligung an Diskussionen, Präsentationen, „teaching experiments“	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 80%;">Präsenzzeit</td> <td style="text-align: right;">45</td> </tr> <tr> <td>Vor- und Nachbereitung</td> <td style="text-align: right;">75</td> </tr> <tr> <td>Prüfungsvorbereitung und Prüfung</td> <td style="text-align: right;">30</td> </tr> </table>	Präsenzzeit	45	Vor- und Nachbereitung	75	Prüfungsvorbereitung und Prüfung	30
Präsenzzeit	45								
Vor- und Nachbereitung	75								
Prüfungsvorbereitung und Prüfung	30								
Modulprüfung:		Hausarbeit (ca. 8 Seiten) oder Präsentation (ca. 15 Minuten) oder Klausur (45 Minuten), ggf. ganz oder teilweise in der Form des Antwort-Wahl-Verfahrens; kann auch in Form einer elektronischen Prüfungsleistung durchgeführt werden							
Veranstaltungssprache:		Deutsch							
Pflicht zur regelmäßigen Teilnahme:		Ja							
Arbeitsaufwand insgesamt:		150 Stunden	5 LP						
Dauer des Moduls:		Ein Semester							
Häufigkeit des Angebots:		Jedes Sommersemester							
Verwendbarkeit:		Masterstudiengang für das Lehramt an Integrierten Sekundarschulen und Gymnasien mit dem Profil Quereinstieg							

² Studien- und Prüfungsordnung der Freien Universität Berlin für den Masterstudiengang für das Lehramt an Integrierten Sekundarschulen und Gymnasien mit dem Profil Quereinstieg



Modul: Schulpraktische Studien im Unterrichtsfach Informatik Fach 1³

Modul: Schulpraktische Studien im Unterrichtsfach Informatik – Fach 1
Hochschule/Fachbereich: Freie Universität Berlin/FB Mathematik und Informatik
Modulverantwortliche/r: Professur für Didaktik der Informatik
Zugangsvoraussetzungen: Keine
Qualifikationsziele: <p>Die Studentinnen und Studenten kennen zentrale Konzepte und Bedingungen für die Planung von Informatikunterricht in verschiedenen Schulformen und können diese aufeinander beziehen. Sie treffen dementsprechend begründete Planungsentscheidungen und reflektieren diese. Bei der Gestaltung von Lern- und Bildungsprozessen berücksichtigen die Studentinnen und Studenten individuelle Lernvoraussetzungen ebenso wie inklusionspädagogische Prinzipien und die Kerndimensionen von Diversity (unter anderem: Geschlecht, sexuelle Orientierung, sozialer Status, Migration, Förderbedarf). Sie wissen um die Bedeutung von Selbsttätigkeit und Eigenverantwortlichkeit der Schülerinnen und Schüler beim Lernen von Informatik. Ihre Unterrichtsplanungen zielen auf die Schaffung derartiger Lernumgebungen. Aufgabenstellungen konzipieren und formulieren sie kriteriengeleitet, schulformbezogen und adressatengerecht. Dabei erkennen sie Benachteiligungen und Förderbedarfe und reagieren mit didaktischen Angeboten. Sie können Lernstände erheben und fachliches Lernen beurteilen sowie diese Leistungsüberprüfungen als konstruktive Rückmeldung über die eigene Unterrichtstätigkeit nutzen. Intendierte und nicht intendierte Effekte von eigenem und fremdem Fachunterricht können die Studentinnen und Studenten reflektieren. Verlauf und Ergebnisse des eigenen Unterrichts analysieren und beurteilen sie mit Mitteln der Selbst- und Fremdevaluation. Auf dieser Basis können sie Alternativen entwerfen und ihren Informatikunterricht weiterentwickeln. Die in diesem Zusammenhang erworbenen Selbstregulationskompetenzen befähigen sie dazu, persönliche Ressourcen und Ziele zu reflektieren und weiterzuentwickeln. Die Studentinnen und Studenten verfügen über Kommunikationskompetenzen und können fachliche Fragen mit Lernenden, Eltern, Kolleginnen und Kollegen diskutieren. Die Studentinnen und Studenten können konkrete Sprachhandlungen des Informatikunterrichts schulformbezogen benennen, analysieren und zum Gegenstand Informatikdidaktischer Reflexion machen. Sie wenden sprachbildende/DaZ Prinzipien des Informatikunterrichts in Unterrichtsentwürfen an.</p>
Inhalte: <p>Bei der Planung von Informatikunterricht werden unter anderem Kompetenzbereiche und Basiskonzepte der Bildungsstandards schulformbezogen, curriculare Vorgaben, Lernvoraussetzungen der Schülerinnen und Schüler, Sachanalyse und fachspezifische Strukturierung, schulformbezogen didaktische und methodische Überlegungen, Kompetenzen/Unterrichtsziele, Impulsgebung berücksichtigt. Bei der Durchführung und Reflexion von Informatikunterricht stehen schulformbezogen fachspezifische Aspekte der Unterrichtsorganisation, Verhältnis von Planung und Durchführung, Lernklima und Lernentwicklung, Lehrerverhalten und Lehrersprache, Angemessenheit der Lernumgebung und Methodik, Bewertung der Lernentwicklung, kritische Reflexion der eigenen fachlichen Voraussetzungen, Entwicklung von begründeten Alternativen im Mittelpunkt.</p>

³ Studien- und Prüfungsordnung für den Masterstudiengang für das Lehramt an Integrierten Sekundarschulen und Gymnasien der Freien Universität Berlin



Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Seminar (Vorbereitung)	2	eigenständige Lektüre, aktive Beteiligung am Seminargespräch, Kurzreferat, Erstellen eines Handouts, Analyse und Erarbeitung von Lehr-Lernmaterialien, Unterrichtsentwurf	Präsenzzeit S 30 Vor- und Nachbereitung S 45
Schulpraktikum	4	Hospitation, Planung, Durchführung und Reflexion angeleiteten Unterrichts, Unterrichtsvor- und -nachbesprechungen mit Mentorinnen, Mentoren, Dozentinnen, Dozenten sowie Fachberaterinnen und Fachberatern, sonstige Aufgaben	Präsenzzeit in der Schule SP einschließlich Vor- und Nachbereitung in der Schule 160
Seminar (Begleitung und Reflexion am „Uni-Tag“ und/oder zur Nachbereitung im Block am Ende des Praxissemesters)	2	eigenständige Lektüre, aktive Beteiligung am Seminargespräch, Kurzreferat, Erstellen eines Handouts, Analyse und Erarbeitung von Lehr-Lernmaterialien, Erstellen, Analysieren und Überarbeiten von Unterrichtsentwürfen, Analysieren und Reflektieren von eigenem und fremdem Unterricht	Präsenzzeit S 30 Vor- und Nachbereitung 45
			Präsenzzeit S 30 Vor- und Nachbereitung S 45 Prüfungsvorbereitung und Prüfung 50
Modulprüfung:		Praktikumsbericht (ca. 40 Seiten inkl. Anhang)	
Modulsprache:		Deutsch	
Pflicht zur regelmäßigen Teilnahme:		Ja	
Arbeitsaufwand insgesamt:		435 Stunden	12 LP
Dauer des Moduls:		Zwei Semester	
Häufigkeit des Angebots:		Seminar (Vorbereitung): Sommersemester, Schulpraktikum und Seminar (Begleitung/Reflexion): Wintersemester (Praxissemester)	
Verwendbarkeit:		Masterstudiengang für das Lehramt an Integrierten Sekundarschulen und Gymnasien	



Modul: Schulpraktische Studien im Unterrichtsfach Informatik Fach 2⁴

Modul: Schulpraktische Studien im Unterrichtsfach Informatik – Fach 2
Hochschule/Fachbereich: Freie Universität Berlin/FB Mathematik und Informatik
Modulverantwortliche/r: Professur für Didaktik der Informatik
Zugangsvoraussetzungen: Keine
Qualifikationsziele: Die Studentinnen und Studenten kennen zentrale Konzepte und Bedingungen für die Planung von Informatikunterricht in verschiedenen Schulformen und können diese aufeinander beziehen. Sie treffen dementsprechend begründete Planungsentscheidungen und reflektieren diese. Bei der Gestaltung von Lern- und Bildungsprozessen berücksichtigen die Studentinnen und Studenten individuelle Lernvoraussetzungen ebenso wie inklusionspädagogische Prinzipien und die Kerndimensionen von Diversity (unter anderem: Geschlecht, sexuelle Orientierung, sozialer Status, Migration, Förderbedarf). Sie wissen um die Bedeutung von Selbsttätigkeit und Eigenverantwortlichkeit der Schülerinnen und Schüler beim Lernen von Informatik. Ihre Unterrichtsplanungen zielen auf die Schaffung derartiger Lernumgebungen. Aufgabenstellungen konzipieren und formulieren sie kriteriengeleitet, schulformbezogen und adressatengerecht. Dabei erkennen sie Benachteiligungen und Förderbedarfe und reagieren mit didaktischen Angeboten. Sie können Lernstände erheben und fachliches Lernen beurteilen sowie diese Leistungsüberprüfungen als konstruktive Rückmeldung über die eigene Unterrichtstätigkeit nutzen. Intendierte und nicht intendierte Effekte von eigenem und fremdem Fachunterricht können die Studentinnen und Studenten reflektieren. Verlauf und Ergebnisse des eigenen Unterrichts analysieren und beurteilen sie mit Mitteln der Selbst- und Fremdevaluation. Auf dieser Basis können sie Alternativen entwerfen und ihren Informatikunterricht weiterentwickeln. Die in diesem Zusammenhang erworbenen Selbstregulationskompetenzen befähigen sie dazu, persönliche Ressourcen und Ziele zu reflektieren und weiterzuentwickeln. Die Studentinnen und Studenten verfügen über Kommunikationskompetenzen und können fachliche Fragen mit Lernenden, Eltern, Kolleginnen und Kollegen diskutieren. Die Studentinnen und Studenten können konkrete Sprachhandlungen des Informatikunterrichts schulformbezogen benennen, analysieren und zum Gegenstand Informatikdidaktischer Reflexion machen. Sie wenden sprachbildende/DaZ Prinzipien des Informatikunterrichts in Unterrichtsentwürfen an.
Inhalte: Bei der Planung von Informatikunterricht werden unter anderem Kompetenzbereiche und Basiskonzepte der Bildungsstandards schulformbezogen, curriculare Vorgaben, Lernvoraussetzungen der Schülerinnen und Schüler, Sachanalyse und fachspezifische Strukturierung, schulformbezogen didaktische und methodische Überlegungen, Kompetenzen/Unterrichtsziele, Impulsgebung berücksichtigt. Bei der Durchführung und Reflexion von Informatikunterricht stehen schulformbezogen fachspezifische Aspekte der Unterrichtsorganisation, Verhältnis von Planung und Durchführung, Lernklima und Lernentwicklung, Lehrerverhalten und Lehrersprache, Angemessenheit der Lernumgebung und Methodik, Bewertung der Lernentwicklung, kritische Reflexion der eigenen fachlichen Voraussetzungen, Entwicklung von begründeten Alternativen im Mittelpunkt.



Lehr- und Lernformen	Präsenzstudium (Semesterwochen- stunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)	
Seminar (Vorbereitung)	2	eigenständige Lektüre, aktive Beteiligung am Semi- nargespräch, Kurzreferat, Erstellen eines Handouts, Analyse und Erarbeitung von Lehr-Lernmaterialien, Unterrichtsentwurf	Präsenzzeit S 30 Vor- und Nachbereitung S 45	
Schulpraktikum	4	P R A X I S S	Hospitation, Planung, Durchführung und Refle- xion angeleiteten Unter- richts, Unterrichtsvor- und -nachbesprechungen mit Mentorinnen, Mentoren, Dozentinnen, Dozenten sowie Fachberaterinnen und Fachberatern, sonstige Aufgaben	Präsenzzeit in der Schule SP einschließlich Vor- und Nachbereitung in der Schule 160
Seminar (Begleitung und Re- flexion am „Uni-Tag“ und/oder zur Nach- bereitung im Block am Ende des Praxis- semesters)	2	S E M E S T E R	eigenständige Lektüre, aktive Beteiligung am Semi- nargespräch, Kurzreferat, Erstellen eines Handouts, Analyse und Erarbeitung von Lehr-Lernmaterialien, Erstellen, Analysieren und Überarbeiten von Unter- richtsentwürfen, Analysieren und Reflektieren von eige- nem und fremdem Unter- richt, Reflexionsleistung zur Praxiserfahrung	Präsenzzeit S 30 Vor- und Nachbereitung S 95
Modulprüfung:	Keine			
Modulsprache:	Deutsch			
Pflicht zur regelmäßigen Teilnahme:	Ja			
Arbeitsaufwand insgesamt:	360 Stunden		12 LP	
Dauer des Moduls:	Zwei Semester			
Häufigkeit des Angebots:	Seminar (Vorbereitung): Sommersemester, Schulpraktikum und Seminar (Begleitung/Reflexion): Wintersemester (Praxissemester)			
Verwendbarkeit:	Masterstudiengang für das Lehramt an Integrierten Sekundarschulen und Gymnasien			



Modul: Bildverarbeitung⁵

Modul: Bildverarbeitung			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentinnen und Dozenten des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten können Bilder, die von Digital- und Videokameras stammen, verbessern und verändern und sie für den weitergehenden Gebrauch durch Menschen oder die Verarbeitung durch Maschinen nutzbar machen. Sie verstehen die grundlegenden Qualitätsbegriffe, auf die es dabei ankommt, und die algorithmischen Techniken, die dabei verwendet werden.			
Inhalte: Es werden grundlegende Bildverarbeitungstechniken behandelt. Diese umfassen Farbkorrekturen von Bildern, Fouriertransformation, Glätten, Schärfen, Kantendetektion, Aufbau von Bildpyramiden, ScaleSpace-Theory sowie grundlegende Verfahren zur Mustererkennung, wie z. B. die Hough-Transformation.			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	2	Bearbeitung der Übungsaufgaben	Präsenzzeit V 30 Vor- und Nachbereitung V 30 Präsenzzeit Ü 30 Vor- und Nachbereitung Ü 30 Prüfungsvorbereitung und Prüfung 30
Übung	2		
Modulprüfung:		Klausur (90 Minuten), die Klausur kann auch in Form einer elektronischen Prüfungsleistung (90 Minuten) durchgeführt werden, oder mündliche Prüfung (20 bis 25 Minuten)	
Veranstaltungssprache:		Deutsch (ggf. Englisch)	
Pflicht zu regelmäßiger Teilnahme:		Teilnahme wird empfohlen	
Arbeitszeitaufwand insgesamt:		150 Stunden	5 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Zweijährlich	
Verwendbarkeit:		Masterstudiengang Informatik	

⁵ Studien- und Prüfungsordnung für den Masterstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Computergrafik⁶

Modul: Computergrafik			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentinnen und Dozenten des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Studierende kennen die auf dem gesamten Weg von der Modellierung zur graphischen Darstellung auftretenden Problemstellungen. Sie wissen exemplarisch, wie diese Fragen in den gängigen Systemen in Hardware oder in Software gelöst werden können und sie verstehen die geometrischen und physikalischen Grundlagen, die nötig sind, um mit fortgeschrittenen Computergrafik-Systemen umzugehen.			
Inhalte: Mathematische Grundlagen der Computergrafik, Darstellung von 3-D-Szenen im Rechner, geometrische Transformationen, Projektionen auf die Bildebene, Bestimmung sichtbarer Flächen, Beleuchtungsmodelle, Ray-Tracing, Radiosity, Animation.			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	4	Bearbeitung der Übungsaufgaben	Präsenzzeit V 60
			Vor- und Nachbereitung V 60
Übung	2		Präsenzzeit Ü 30
			Vor- und Nachbereitung Ü 90
			Prüfungsvorbereitung und Prüfung 60
Modulprüfung:		Klausur (90 Minuten), die Klausur kann auch in Form einer elektronischen Prüfungsleistung (90 Minuten) durchgeführt werden.	
Veranstaltungssprache:		Deutsch (ggf. Englisch)	
Pflicht zu regelmäßiger Teilnahme:		Teilnahme wird empfohlen	
Arbeitszeitaufwand insgesamt:		300 Stunden	10 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Zweijährlich	
Verwendbarkeit:		Masterstudiengang Informatik	

⁶Studien- und Prüfungsordnung für den Masterstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Computer-Vision⁷

Modul: Computer-Vision			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentinnen und Dozenten des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten kennen Methoden des Computersehens auf einem aktuellen Stand und können ein Computersystem für die Erkennung von Objekten und Umgebungen programmieren (z. B. für den Betrieb eines Roboters).			
Inhalte: Computer-Vision arbeitet im Gegensatz zur reinen Bildverarbeitung mit einer Folge von Bildern und versucht daraus Objekte zu erkennen und ein räumliches Modell zu konstruieren. Die Teilnehmerinnen und Teilnehmer werden anhand von aktueller Literatur mit dem gegenwärtigen Stand der Forschung auf diesem Gebiet vertrautgemacht.			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	2	Bearbeitung der Übungsaufgaben	Präsenzzeit V 30
			Vor- und Nachbereitung V 30
Übung	2		Präsenzzeit Ü 30
			Vor- und Nachbereitung Ü 30
			Prüfungsvorbereitung und Prüfung 30
Modulprüfung:		Klausur (90 Minuten), die Klausur kann auch in Form einer elektronischen Prüfungsleistung (90 Minuten) durchgeführt werden, oder mündliche Prüfung (20 bis 25 Minuten)	
Veranstaltungssprache:		Deutsch (ggf. Englisch)	
Pflicht zu regelmäßiger Teilnahme:		Teilnahme wird empfohlen	
Arbeitszeitaufwand insgesamt:		150 Stunden	5 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Zweijährlich	
Verwendbarkeit:		Masterstudiengang Informatik	

⁷ Studien- und Prüfungsordnung für den Masterstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Datenbanktechnologie⁸

Modul: Datenbanktechnologie			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentinnen und Dozenten des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten beherrschen aktuelle, technische Verfahren zur effizienten und sicheren Verwaltung von Daten und können fehlertolerante, effiziente Datenbanksysteme entwickeln und deren Qualität beurteilen.			
Inhalte: Das Studium beinhaltet alle technischen Fragen, die sich im Zusammenhang mit der Implementierung von Datenverwaltungssystemen stellen. Dazu gehören Zugriffstechniken und Anfrageoptimierung, die Realisierung von Transaktionen, insbesondere Synchronisationsverfahren, die technische Maßnahmen, die Datenbanksysteme fehlertolerant machen. Neben den in relationalen Systemen verwendeten Techniken werden Verfahren zur effizienten Verwaltung andersartiger großer Datenbestände, insbesondere von XML-Dokumenten, behandelt. Ein Schwerpunkt der Veranstaltung ist die korrekte Implementierung transaktionaler Garantien in Datenverwaltungssystemen.			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	2	Bearbeitung der Übungsaufgaben	Präsenzzeit V 30
			Vor- und Nachbereitung V 30
Übung	2		Präsenzzeit Ü 30
			Vor- und Nachbereitung Ü 30
			Prüfungsvorbereitung und Prüfung 30
Modulprüfung:		Klausur (90 Minuten), die Klausur kann auch in Form einer elektronischen Prüfungsleistung (90 Minuten) durchgeführt werden, oder mündliche Prüfung (20 bis 25 Minuten)	
Veranstaltungssprache:		Deutsch (ggf. Englisch)	
Pflicht zu regelmäßiger Teilnahme:		Teilnahme wird empfohlen	
Arbeitszeitaufwand insgesamt:		150 Stunden	5 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Jedes Wintersemester	
Verwendbarkeit:		Masterstudiengang Informatik	

⁸ Studien- und Prüfungsordnung für den Masterstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin

Modul: Grundlagen des Softwaretestens⁹

Modul: Grundlagen des Softwaretestens			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentinnen und Dozenten des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten verstehen die Grundlagen des Softwaretestens und die Rolle des Testens im gesamten Software-Lebenszyklus. Sie kennen die Stufen und Typen von Softwaretests. Sie können Tests nach dem Stand der Technik entwerfen, und sie können ihre Kenntnisse bei der Abwicklung von Tests anwenden.			
Inhalte: Das Testen nimmt einen immer größeren Stellenwert in der Entwicklung und Qualitätssicherung software-basierter Systeme ein. Diese Vorlesung wird Basiskonzepte des Software-testens erläutern und praxisrelevante Methoden zum Testmanagement, zum Testentwurf, zur Testspezifikation, Testgenerierung und Testbewertung vermitteln. Folgende Themenblöcke werden behandelt: <ul style="list-style-type: none"> • Grundlagen des Softwaretestens • Testen im Softwarelebenszyklus • Statischer Test • Dynamischer Test • Testfallentwurfsverfahren • Testmanagement • Testwerkzeuge Die Vorlesung basiert auf dem ISTQB® (International Software Testing Qualification Board, www.istqb.org) Certified Tester Programm, einem weltweit anerkannten, standardisierten Aus- und Weiterbildungsschema für Software-Tester. Die Vorlesung vermittelt den Stoff des ISTQB Software Tester Foundation Level und weiterführende aktuelle Testmethoden und -techniken. Im Anschluss kann daher neben der Prüfung zur Vorlesung eine Prüfung zum Zertifikat Software Tester Foundation Level abgelegt werden. Dieses Zertifikat wird mittlerweile in vielen Stellenausschreibungen nachgefragt.			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	2	Bearbeitung der Übungsblätter	Präsenzzeit V 30 Vor- und Nachbereitung V 30
Übung	2	Zwei mündliche Präsentationen der Lösung jeweils einer Übungsaufgabe in der Übung	Präsenzzeit Ü 30 Vor- und Nachbereitung Ü 30 Prüfungsvorbereitung und Prüfung 30
Modulprüfung:		Klausur (90 Minuten), die Klausur kann auch in Form einer elektronischen Prüfungsleistung (90 Minuten) durchgeführt werden, oder mündliche Prüfung (20 bis 25 Minuten)	
Veranstaltungssprache:		Deutsch (ggf. Englisch)	
Pflicht zu regelmäßiger Teilnahme:		Teilnahme wird empfohlen	
Arbeitszeitaufwand insgesamt:		150 Stunden	5 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Jedes Sommersemester	
Verwendbarkeit:		Masterstudiengang Informatik	

⁹ Studien- und Prüfungsordnung für den Masterstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Künstliche Intelligenz¹⁰

Modul: Künstliche Intelligenz			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentinnen und Dozenten des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten beherrschen die grundlegenden Techniken, Heuristiken und Algorithmen auf dem Gebiet der Künstlichen Intelligenz und können sie sowohl für symbolische als auch für Mustererkennungsprobleme anwenden.			
Inhalte: Suchverfahren für die Lösung kombinatorischer Aufgaben, Prädikatenlogik und ihre Mechanisierung, Resolution und Theorembeweise, Wissensbasierte- und Expertensysteme, Diffuse Logik, Mensch-Maschinen-Schnittstellen, Mustererkennung insbesondere für Handschrift und für gesprochene Sprache			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	2	Bearbeitung der Übungsblätter	Präsenzzeit V 30 Vor- und Nachbereitung V 30
Übung	2	Zwei mündliche Präsentationen der Lösung jeweils einer Übungsaufgabe in der Übung	Präsenzzeit Ü 30 Vor- und Nachbereitung Ü 30 Prüfungsvorbereitung und Prüfung 30
Modulprüfung:		Klausur (90 Minuten), die Klausur kann auch in Form einer elektronischen Prüfungsleistung (90 Minuten) durchgeführt werden, oder mündliche Prüfung (20 bis 25 Minuten)	
Veranstaltungssprache:		Deutsch (ggf. Englisch)	
Pflicht zu regelmäßiger Teilnahme:		Teilnahme wird empfohlen	
Arbeitszeitaufwand insgesamt:		150 Stunden	5 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Jedes Sommersemester	
Verwendbarkeit:		Masterstudiengang Informatik	

¹⁰ Studien- und Prüfungsordnung für den Masterstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Medizinische Bildverarbeitung¹¹

Modul: Medizinische Bildverarbeitung			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentinnen und Dozenten des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten können die Qualität und Eigenschaften medizinischen Bildmaterials beurteilen. Sie kennen spezielle Eigenschaften medizinischen Bildmaterials, die in der Anwendung von Algorithmen zu berücksichtigen sind und können eine problembezogene Auswahl geeigneter Bildverarbeitungsalgorithmen treffen und diese zu Gesamtlösungen verknüpfen. Sie beherrschen Methoden zu Bildverbesserung, Registrierung, Segmentierung und Klassifikation und können diese selbstständig anwenden. Sie können die Güte von Bildverarbeitungsalgorithmen sicher beurteilen.			
Inhalte: Einführung in die medizinische Bildverarbeitung, Zielsetzungen digitaler Bildverarbeitung in der Medizin, Extraktion von Informationen aus Bilddaten, Objekterkennung (Unterstützung der Wahrnehmung von Bildinformation, Bildkontrast, Filterung, Texturerkennung, Segmentierung) und Probleme in der medizinischen Praxis, relative Lage von Bildern (Alignment, 3D-Bildrekonstruktion), Bewegtbildern und Objektverfolgung. Anwendungsbeispiele: behandelt werden manuelle, interaktive und automatische Methoden (intensitäts- und modellbasiert) auf der Grundlage medizinischen Bildmaterials.			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	2	Bearbeitung der Übungsaufgaben	Präsenzzeit V 30
			Vor- und Nachbereitung V 45
Übung	1		Präsenzzeit Ü 15
			Vor- und Nachbereitung Ü 30
			Prüfungsvorbereitung und Prüfung 30
Modulprüfung:		Klausur (90 Minuten), die Klausur kann auch in Form einer elektronischen Prüfungsleistung (90 Minuten) durchgeführt werden, oder mündliche Prüfung (20 bis 25 Minuten)	
Veranstaltungssprache:		Deutsch (ggf. Englisch)	
Pflicht zu regelmäßiger Teilnahme:		Teilnahme wird empfohlen	
Arbeitszeitaufwand insgesamt:		150 Stunden	5 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Jedes Semester	
Verwendbarkeit:		Masterstudiengang Informatik	

¹¹ Studien- und Prüfungsordnung für den Masterstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Modellgetriebene Softwareentwicklung¹²

Modul: Modellgetriebene Softwareentwicklung			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentinnen und Dozenten des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten kennen Konzepte, Methoden und Werkzeuge der modellgetriebenen Softwareentwicklung. Sie können die dynamischen und statischen Aspekte von softwareintensiven Systemen anhand verschiedener Modellierungssprachen beschreiben. Sie verstehen den Einsatz und die Anwendungsgrenzen von modellgetriebener Softwareentwicklung im Rahmen des generellen Softwareentwicklungsprozesses.			
Inhalte: Auf den bereits erworbenen Kenntnissen der UML aufbauend, werden zuerst grundlegende Konzepte der Metamodellierung betrachtet. Anschließend wird der Bereich der domänenspezifischen Sprachen (DSL – Domain Specific Languages) behandelt. Es wird der Entwurf und die Implementierung von DSLs im Rahmen des gesamten Softwareentwicklungsprozesses betrachtet, angefangen von der Motivation über Konzeption bis hin zu Kodengeneration und Ausführung. Auf der Ebene der Modelle werden Ansätze der Modellanalyse wie Model Checking und die Transformation von Modellen behandelt. Dabei werden sowohl Modell-zu-Modell-Transformationen, wie die Abbildung eines plattformunabhängigen Modells auf eine konkrete Ausführungsplattform oder verhaltensneutrale Refactorings von Modellen als auch die Modell-zu-Text-Transformationen, wie sie beispielsweise für die Kodenerzeugung verwendet werden, betrachtet. Der letzte thematische Block des Moduls beschäftigt sich mit der Verwendung von Modellen zur Laufzeit. Die Interpretation von Verhaltensmodellen wird genauer behandelt und der Zusammenhang zwischen Strukturmodellen und dynamischen Komponentensystemen wird näher beleuchtet. Die Übungen werden parallel durchgeführt und machen den theoretisch vermittelten Stoff durch praktische Anwendung der gelernten Konzepte und Ansätze besser verständlich.			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	2	Bearbeitung der Übungsblätter Zwei mündliche Präsentationen der Lösung jeweils einer Übungsaufgabe in der Übung	Präsenzzeit V 30 Vor- und Nachbereitung V 30 Präsenzzeit Ü 30 Vor- und Nachbereitung Ü 30
Übung	2		Prüfungsvorbereitung und Prüfung 30
Modulprüfung:		Klausur (90 Minuten), die Klausur kann auch in Form einer elektronischen Prüfungsleistung (90 Minuten) durchgeführt werden, oder mündliche Prüfung (20 bis 25 Minuten)	
Veranstaltungssprache:		Deutsch (ggf. Englisch)	
Pflicht zu regelmäßiger Teilnahme:		Teilnahme wird empfohlen	
Arbeitszeitaufwand insgesamt:		150 Stunden	5 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Jedes Wintersemester	
Verwendbarkeit:		Masterstudiengang Informatik	

¹² Studien- und Prüfungsordnung für den Masterstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Netzbasierte Informationssysteme¹³

Modul: Netzbasierte Informationssysteme			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentinnen und Dozenten des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten kennen die Grundlagen von Technologien, die zum Bau netzbasierter Informationssysteme notwendig sind und verstehen die wichtigsten Mechanismen und ihre Zusammenhänge. Sie sind in der Lage, diese einzuordnen und geeignet darzustellen.			
Inhalte: Netzbasierte Informationssysteme stellen mit der Verbreitung des Web im weltweiten Maßstab Informationen bereit. Die Vorlesung vermittelt Kenntnisse über die wichtigsten Technologien, Probleme und Lösungsansätze solcher Systeme. Die Veranstaltung gliedert sich in vier Bereiche (in Klammern behandelte Technologien und Konzepte): <ul style="list-style-type: none"> • Das Web: Wie sind Inhalte repräsentiert (HTML/XML), wie findet man sie (Crawling, Deep Web), wie kann man darauf zugreifen (Internet-Protokolle)? • Web Suche: Information Retrieval für das Web, Indexing, Multimedia Indexing, Collaborative Filtering, Nutzung der Web-Struktur bei der Suche (PageRank, HITS), Metasuchmaschinen • Betrieb, Ausführung und Darstellung von Web-Sites: Nutzung und Nutzer von Web-Sites, Betriebsaspekte sehr großer Dienste, Server- und Clientseitige Ausführung, Caching in Web, Clientseitige Darstellung, Mehrsprachigkeit im Web • Semantic Web: Technologien und Anwendungen Neben dem Vorlesungsteil werden im Übungsteil ergänzende Themen, beispielsweise relevante Internet- und Web-Standarddokumente in Referaten behandelt.			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	2	Bearbeitung der Übungsblätter	Präsenzzeit V 30 Vor- und Nachbereitung V 30
Übung	2	Zwei mündliche Präsentationen der Lösung jeweils einer Übungsaufgabe in der Übung	Präsenzzeit Ü 30 Vor- und Nachbereitung Ü 30 Prüfungsvorbereitung und Prüfung 30
Modulprüfung:		Klausur (90 Minuten), die Klausur kann auch in Form einer elektronischen Prüfungsleistung (90 Minuten) durchgeführt werden, oder mündliche Prüfung (20 bis 25 Minuten)	
Veranstaltungssprache:		Deutsch (ggf. Englisch)	
Pflicht zu regelmäßiger Teilnahme:		Teilnahme wird empfohlen	
Arbeitszeitaufwand insgesamt:		150 Stunden	5 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Zweijährlich	
Verwendbarkeit:		Masterstudiengang Informatik	

¹³ Studien- und Prüfungsordnung für den Masterstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Rechnersicherheit¹⁴

Modul: Rechnersicherheit			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentinnen und Dozenten des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele:			
Die Studentinnen und Studenten sind in der Lage,			
<ul style="list-style-type: none"> • typische Angriffe auf Daten- und IT-Sicherheit zu benennen und ihr Schadenspotential anwendungsbezogen einzuschätzen, • Prinzipien, Methoden und Mechanismen zum Schutz von Systemen zu benennen und ihre Einsatzbereiche zu beschreiben, • in Kenntnis potentieller Sicherheitslücken Systeme hinsichtlich ihrer Sicherheitseigenschaften zu analysieren, • bei der Software-Entwicklung Sicherheitsbelange bereits bei der Anforderungsdefinition und anschließend während des gesamten Entwicklungsprozesses zu berücksichtigen und • betriebliche Sicherheitsrichtlinien sowie Datenschutzrichtlinien technisch umzusetzen; sie kennen die Bestimmungen des Datenschutzrechts. 			
Inhalte:			
<p>Grundbegriffe: Schutzziele, Sicherheitsmechanismen, Umsetzung von Sicherheitsanforderungen, Systemsicherheit versus Netzsicherheit. Gesellschaftlicher Kontext: Historisches, Politisches, Evaluation und Zertifizierung. Typische Angriffe: Trojanische Pferde, Salami-Taktik, Geheimtüren, Viren, Würmer, Logische Bomben, verdeckte Lecks, Ausnutzung von Software-Qualitätsmängeln (z. B. Pufferüberlauf).</p> <p>Zugangskontrolle: Passwörter, Sicherungskarten, Biometrie. Zugriffsschutz: Speicherschutz, Autorisierung eines Prozesses, Dateischutz, Capabilities, Modellierung, rollenbasierter Zugriffsschutz, Zugriffsschutzstrategien, Zugriffsschutz in Programmiersprachen, Sicherheitsmechanismen in Java, anwendungsorientierte Schutzsysteme (Datenbanken, CORBA). Überwachungssysteme: Auditing, Intrusion Detection.</p> <p>Informationsflusskontrolle: Elemente der Informationstheorie, Informationsfluss zwischen Objekten, Sicherheitsklassen, mehrstufige Sicherheit, flusssichere Programme, Zugriffsschutz und Flusskontrolle (Bell-LaPadula-Modell, Chinese-Wall-Modell). Sicherheitsmechanismen in lokalen Netzen: Zugangskontrolle über Sun NIS, Fernbenutzung (telnet, ssh), Zugriffsschutz in verteilten Dateisystemen. Kryptographie: Grundbegriffe, Transpositionsverschlüsselung, Substitutionsverschlüsselung, Sicherheit von Verschlüsselungsverfahren, Polyalphabetische Substitution, sichere Blockverschlüsselung; asymmetrische Verschlüsselung (knapsack, RSA); Authentizität, digitale Unterschriften, Hash-Codes, DSS. Kryptographische Protokolle: Elementare Protokolle, Schlüsselverwaltung, Diffie-Hellman, Zertifikate, PKI, PGP, Authentisierungsdienste (Kerberos, Sesame). Sichere Endsysteme: Trusted Computing: TCG, TPM, Secure Booting, Pro&Contra, DRM.</p>			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	4	Regelmäßige, schriftliche Bearbeitung der Übungsblätter	Präsenzzeit V 60 Vor- und Nachbereitung V 90 Präsenzzeit Ü 30
Übung	2	Zwei mündliche Präsentationen der Lösung einer Übungsaufgabe	Vor- und Nachbereitung Ü 60 Prüfungsvorbereitung und Prüfung 60
Modulprüfung:		Klausur (90 Minuten) , die Klausur kann auch in Form einer elektronischen Prüfungsleistung (90 Minuten) durchgeführt werden, oder mündliche Prüfung (20 bis 25 Minuten)	
Veranstaltungssprache:		Deutsch (ggf. Englisch)	
Pflicht zu regelmäßiger Teilnahme:		Teilnahme wird empfohlen	
Arbeitszeitaufwand insgesamt:		300 Stunden	10 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Jedes Wintersemester	
Verwendbarkeit:		Masterstudiengang Informatik	

¹⁴ Studien- und Prüfungsordnung für den Masterstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Übersetzerbau¹⁵

Modul: Übersetzerbau			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentinnen und Dozenten des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten kennen die wesentlichen Phasen eines Übersetzers und beherrschen die allgemeinen Techniken für jede Phase. Sie können die Techniken des Übersetzerbaus auch in anderen Anwendungsbereichen einsetzen.			
Inhalte: Ein Übersetzer ist ein Programm, das Programme einer höheren Programmiersprache in eine andere Programmiersprache (im Allgemeinen Maschinensprache) überführt. In der Regel erfolgt die Übersetzung in mehreren Phasen, wovon die wichtigsten die lexikalische Analyse, die Syntaxanalyse, die semantische Analyse und die Codeerzeugung sind. Mit Hilfe der lexikalischen und syntaktischen Analyse wird das Quellprogramm in eine computergerechte Repräsentation überführt (abstrakter Syntaxbaum). Diese Repräsentation wird dann als Ausgangspunkt für Optimierungen und die Codeerzeugung verwendet. Die hier vorgestellten Verfahren finden an vielen Stellen in der Informatik Anwendung. Deshalb ist dieses Thema auch für solche Hörer von Interesse, die nie vorhaben, einen Übersetzer zu schreiben.			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	4	Regelmäßige, schriftliche Bearbeitung der Übungsblätter	Präsenzzeit V 60 Vor- und Nachbereitung V 60 Präsenzzeit Ü 30
Übung	2	Zwei mündliche Präsentationen der Lösung einer Übungsaufgabe	Vor- und Nachbereitung Ü 90 Prüfungsvorbereitung und Prüfung 60
Modulprüfung:		Klausur (90 Minuten), die Klausur kann auch in Form einer elektronischen Prüfungsleistung (90 Minuten) durchgeführt werden, oder mündliche Prüfung (20 bis 25 Minuten)	
Veranstaltungssprache:		Deutsch (ggf. Englisch)	
Pflicht zu regelmäßiger Teilnahme:		Teilnahme wird empfohlen	
Arbeitszeitaufwand insgesamt:		300 Stunden	10 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Zweijährlich	
Verwendbarkeit:		Masterstudiengang Informatik	

¹⁵ Studien- und Prüfungsordnung für den Masterstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: XML - Technologien¹⁶

Modul: XML-Technologien			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentinnen und Dozenten des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten besitzen vertiefte Kenntnis grundlegender XML-Technologien. So können sie insbesondere deren Stellenwert für das Web der Zukunft aber auch deren Beschränkungen einschätzen.			
Inhalte: Die Extensible Markup Language (XML) ist die neue Sprache des Webs. Sie wird zwar HTML nicht ersetzen, jedoch in einem wichtigen Bereich ergänzen: Während HTML für die Präsentation von elektronischen Dokumenten entwickelt wurde (Mensch-Maschine-Kommunikation), ist XML insbesondere für den Austausch von Daten zwischen Computern geeignet. XML erlaubt dabei die Definition von speziellen Datenaustauschformaten (Standards) sowie die einfache Kombination und Erweiterung solcher Standards. Zusammen mit einer breiten Unterstützung der Software-Industrie ermöglicht dies eine schnelle Verbreitung von XML im Web. Anwendungen von XML findet man heute u. a. in der .NET-Architektur von Microsoft und im E-Business. Es werden folgende Themen behandelt: <ul style="list-style-type: none"> – Ursprünge von XML – Strukturierung von Inhalten mittels XML – Namensräume – Beschreibung von Dokumenten und Daten (DTD- und XML-Schema) – Verarbeitung von XML-Daten (DOM- und SAX-Parser) – Transformation von Dokumenten (XSLT) – XML und Datenbanken – Web Services (SOAP, WSDL) – Semantic Web (RDF, RDFS) Es wird an mittelgroßen Beispielen gezeigt, wie diese Technologien sinnvoll eingesetzt werden können. Gleichzeitig wird das in der Vorlesung erworbene Wissen über die entsprechenden Standards vertieft.			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	2	Regelmäßige, schriftliche Bearbeitung der Übungsblätter	Präsenzzeit V 30 Vor- und Nachbereitung V 30 Präsenzzeit Ü 30
Übung	2	Zwei mündliche Präsentationen der Lösung einer Übungsaufgabe	Vor- und Nachbereitung Ü 30 Prüfungsvorbereitung und Prüfung 30
Modulprüfung:		Klausur (90 Minuten), die Klausur kann auch in Form einer elektronischen Prüfungsleistung (90 Minuten) durchgeführt werden, oder mündliche Prüfung (20 bis 25 Minuten)	
Veranstaltungssprache:		Deutsch (ggf. Englisch)	
Pflicht zu regelmäßiger Teilnahme:		Teilnahme wird empfohlen	
Arbeitszeitaufwand insgesamt:		150 Stunden	5 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Jedes Sommersemester	
Verwendbarkeit:		Masterstudiengang Informatik	

¹⁶ Studien- und Prüfungsordnung für den Masterstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Praktiken professioneller Softwareentwicklung¹⁷

Modul: Praktiken professioneller Softwareentwicklung			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentinnen und Dozenten des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten kennen und verstehen verschiedene Praktiken und können deren Grundgedanken und Zwecke erklären. Sie besitzen praktische Fertigkeiten in der Anwendung dieser Praktiken und können beurteilen, wann und in welchem Grad der Einsatz welcher dieser Praktiken sinnvoll ist.			
Inhalte: Thema sind Entwicklungspraktiken: Konkrete Ausprägungen von allgemeinen Prinzipien der Softwaretechnik in Methoden und Methodenelemente, die alle Aufgabenfelder der Erst- und Fortentwicklung von Software betreffen können (z. B. Anforderungsbestimmung, Spezifikation, Projektplanung, Projektsteuerung und -koordination, Softwareentwurf, Implementierung, Optimierung, Dokumentation, Test, Programmverstehen, Reengineering, Qualitätsmanagement, Betrieb). Die Praktiken können zu eher plangetriebenen, eher agilen oder beiden Entwicklungsstilen passen. Es wird eine Auswahl solcher Praktiken vorgestellt, gemeinsam diskutiert und dann ausprobiert, eingeübt und kritisiert.			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Methodenkurs	2	Teilnahme an der Diskussion	Präsenzzeit MK 30 Vor- und Nachbereitung MK 30
Praxisseminar	2	Präsentation der eig. Arbeiten und Ergebnisse zum Thema der jeweiligen Hausaufgabe	Präsenzzeit PS 30 Vor- und Nachbereitung PS 60
Modulprüfung:		Keine	
Veranstaltungssprache:		Deutsch	
Pflicht zu regelmäßiger Teilnahme:		Ja	
Arbeitszeitaufwand insgesamt:		150 Stunden	5 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Zweijährlich im Wintersemester	
Verwendbarkeit:		Masterstudiengang Informatik	

¹⁷ Studien- und Prüfungsordnung für den Masterstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Semantik von Programmiersprachen¹⁸

Modul: Semantik von Programmiersprachen			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentinnen und Dozenten des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten formalisieren informelle Beschreibungen programmiersprachlicher Konzepte in geeigneter Weise und können sicher mit solchen Formalisierungen arbeiten.			
Inhalte: Das Modul vermittelt Techniken zur Formalisierung der Semantik (Bedeutungsinhalte) von Programmiersprachen. Zunächst werden unterschiedliche Formalisierungsansätze (die operationelle, denotationelle und axiomatische Semantik) vorgestellt und diskutiert. Anschließend wird die mathematische Theorie der semantischen Bereiche behandelt, die bei der denotationellen Methode Anwendung findet. Danach wird schrittweise eine umfassende, imperative Programmiersprache entwickelt und die Semantik der einzelnen Sprachelemente denotationell spezifiziert. Dabei wird die Fortsetzungstechnik (continuation semantics) systematisch erklärt und verwendet. Schließlich wird auf die Anwendung dieser Techniken eingegangen, insbesondere im Rahmen des Compilerbaus und als Grundlage zur Entwicklung funktionaler Programmiersprachen. Dabei wird die besondere Rolle der Verifikation von Programmeigenschaften und der semantikerhaltenden Transformationen hervorgehoben.			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	2	Regelmäßige, schriftliche Bearbeitung der Übungsblätter	Präsenzzeit V 30 Vor- und Nachbereitung V 30
Übung	2	Zwei mündliche Präsentationen der Lösung jeweils einer Übungsaufgabe in der Übung	Präsenzzeit Ü 30 Vor- und Nachbereitung Ü 30 Prüfungsvorbereitung und Prüfung 30
Modulprüfung:		Klausur (90 Minuten), die Klausur kann auch in Form einer elektronischen Prüfungsleistung (90 Minuten) durchgeführt werden, oder mündliche Prüfung (20 bis 25 Minuten)	
Veranstaltungssprache:		Deutsch (ggf. Englisch)	
Pflicht zu regelmäßiger Teilnahme:		Teilnahme wird empfohlen	
Arbeitszeitaufwand insgesamt:		150 Stunden	5 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Zweijährlich	
Verwendbarkeit:		Masterstudiengang Informatik	

¹⁸ Studien- und Prüfungsordnung für den Masterstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Betriebssysteme¹⁹

Modul: Betriebssysteme			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentinnen und Dozenten des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten können Prinzipien, Architektur und Funktionsweise von Betriebssystemen beschreiben und an Betriebssystemen, deren Quellcode vorliegt, Änderungen von mittlerer Komplexität vornehmen. Sie sind in der Lage, typische Dienste, wie sie in heutigen Betriebssystemen an der Systemschnittstelle angeboten werden, für die Entwicklung von Systemsoftware sachgerecht einzusetzen und die Einsatzmöglichkeiten von Betriebssystemen für verschiedene Anwendungsbereiche einzuschätzen. Sie kennen aktuelle Forschungstendenzen und können die Entwicklungstendenzen bei Betriebssystemen einschätzen.			
Inhalte: Einführung: Betriebsarten, Betriebsmittelverwaltung, Historisches, Architektur. Systemdienste: Prozessverwaltung, Adressraumverwaltung, Ein-/Ausgabesystem, Interprozesskommunikation, Dateiverwaltung. Prozessverwaltung: Prozessdeskriptor, Prozessumschaltung, Ablaufsteuerung, Synchronisation, Unterbrechungsbehandlung, Kommunikation. Gerätetreiber: Aufgaben, Einbettung, Auftragspufferung, Fehlerbehandlung, Auftragssteuerung. Speicherverwaltung: Adressraumverwaltung, Prozessumlagerung, Segmentierung, Virtueller Speicher, Segmentierte Prozesse im virtuellen Speicher. Dateiverwaltung: Schnittstelle des Dateisystems, Darstellung der Dateien auf Platten, Implementierung der Dateiverwaltung (Blockpuffer, Deskriptorpuffer), Zugriffsschutz, Dateien als Segmente, persistenter virtueller Speicher. Ein-/Ausgabe: Gerätebenutzung, asynchrone serielle Schnittstellen, Graphikbildschirm. Verteilte Betriebssysteme: Verteilter virtueller Speicher, verteilte Dateisysteme, mobile Prozesse. Stand der Kunst: ausgewählte Beispiele aus der aktuellen Forschung.			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	4	Bearbeitung der Übungsblätter	Präsenzzeit V 60 Vor- und Nachbereitung V 60
Übung	2	Zwei mündliche Präsentationen der Lösung jeweils einer Übungsaufgabe in der Übung	Präsenzzeit Ü 30 Vor- und Nachbereitung Ü 90 Prüfungsvorbereitung und Prüfung 60
Modulprüfung:		Klausur (90 Minuten), die Klausur kann auch in Form einer elektronischen Prüfungsleistung (90 Minuten) durchgeführt werden, oder mündliche Prüfung (20 bis 25 Minuten)	
Veranstaltungssprache:		Deutsch (ggf. Englisch)	
Pflicht zu regelmäßiger Teilnahme:		Teilnahme wird empfohlen	
Arbeitszeitaufwand insgesamt:		300 Stunden	10 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Zweijährlich	
Verwendbarkeit:		Masterstudiengang Informatik	

¹⁹ Studien- und Prüfungsordnung für den Masterstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Robotik²⁰

Modul: Robotik			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentinnen und Dozenten des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten beherrschen die Grundlagen der Robotik und kennen ausgewählte Methoden zur Steuerung von Robotern und zum autonomen Lernen.			
Inhalte: Grundlagen der Robotik, darunter: Computer Vision (lokale, globale), Mechanik, Energiezufuhr, Elektronik, Kommunikation, Steuerung und selbstständiges Lernen der Roboter.			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	2	Regelmäßige Bearbeitung der Übungsaufgaben	Präsenzzeit V 30
			Vor- und Nachbereitung V 30
Übung	2		Präsenzzeit Ü 30
			Vor- und Nachbereitung Ü 30
			Prüfungsvorbereitung und Prüfung 30
Modulprüfung:		Klausur (90 Minuten), die Klausur kann auch in Form einer elektronischen Prüfungsleistung (90 Minuten) durchgeführt werden, oder mündliche Prüfung (20 bis 25 Minuten)	
Veranstaltungssprache:		Deutsch (ggf. Englisch)	
Pflicht zu regelmäßiger Teilnahme:		Teilnahme wird empfohlen	
Arbeitszeitaufwand insgesamt:		150 Stunden	5 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Zweijährlich im Wintersemester (gerades Jahr)	
Verwendbarkeit:		Masterstudiengang Informatik	

²⁰ Studien- und Prüfungsordnung für den Masterstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Objektorientierte Programmierung für Studentinnen und Studenten mit Programmierkenntnissen²¹

Modul: Objektorientierte Programmierung für Studentinnen und Studenten mit Programmierkenntnissen			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentin oder Dozent des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten beherrschen am Ende des Moduls imperative und objektorientierte Programmierkonzepte und deren Anwendbarkeit und können objektorientierte Modelle erstellen. Sie kennen grundlegende Datenstrukturen und Algorithmen und sind in der Lage, abstrakte Datentypen zu spezifizieren und zu implementieren. Sie kennen grundlegende Datenstrukturen und Algorithmen und können die Eigenschaften von kleinen Programmen formal und informell beweisen. Sie können objektorientierte Programme implementieren und dabei Entwurfsmuster problemadäquat einsetzen.			
Inhalte: Dieses Modul richtet sich an Studentinnen und Studenten, deren Einstufungstest Vorkenntnisse in einer objektorientierten Programmiersprache nachweist. Es werden folgende Themen behandelt: <ul style="list-style-type: none"> • Maschinelle Modellierung des Berechenbarkeitsbegriffs (Registernmaschinen) und zentrale Imperative Programmierkonzepte. • Nachweis der Eigenschaften von kleinen Programmen (Hoare-Kalkül und/oder wp-Kalkül). • Objektorientierte Programmierkonzepte (wie Klassen, Objekte, Referenzen, Methoden, Vererbung, polymorphe Typsysteme, Abstrakte Klassen, Schnittstellen, generische Klassendefinitionen, Kapselung, Fehlerbehandlung usw.). • Einfache Datenstrukturen und deren Implementierung unter Verwendung objektorientierter Programmierkonzepte sowie grundlegende Konzepte der Datenabstraktion. • Vertiefte objektorientierte Modellierungstechniken und grundlegende Entwurfsmuster (Iteratoren, Beobachtermuster, Strukturmuster, MVC usw.). • Die Realisierung/Umsetzung der Konzepte werden anhand von modernen, gegenwärtig verwendeten, objektorientierten Programmiersprachen vorgestellt. 			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	3	Einstufungstest, Implementierung einer kleinen Anwendung, Anfertigung und Vorstellen	Präsenzzeit V 45 Vor- und Nachbereitung V 30 Präsenzzeit SPC 30
Seminar am PC	2	85 % der Aufgaben	Vor- und Nachbereitung SPC 105 Prüfungsvorbereitung und Prüfung 30
Modulprüfung:		Klausur (120 Minuten); die Klausur kann auch in Form einer elektronischen Prüfungsleistung (120 Minuten) durchgeführt werden.	
Veranstaltungssprache:		Deutsch	
Pflicht zur regelmäßigen Teilnahme:		Vorlesung: Teilnahme wird empfohlen, Seminar am PC: Ja	
Arbeitszeitaufwand insgesamt:		240 Stunden	8 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Jedes Sommersemester	
Verwendbarkeit:		Bachelorstudiengang Informatik, Bachelorstudiengang Informatik für das Lehramt	

²¹ Studien- und Prüfungsordnung für den Bachelorstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Objektorientierte Programmierung für Studentinnen und Studenten ohne Programmierkenntnissen²²

Modul: Objektorientierte Programmierung für Studentinnen und Studenten ohne Programmierkenntnisse			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentin oder Dozent des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten beherrschen am Ende des Moduls imperative und objektorientierte Programmierkonzepte und deren Anwendbarkeit, eine oder mehrere objektorientierte Programmiersprachen und können objektorientierte Modelle erstellen. Sie kennen grundlegende Datenstrukturen und Algorithmen und sind in der Lage, abstrakte Datentypen zu spezifizieren und zu implementieren. Sie kennen grundlegende Datenstrukturen und Algorithmen und können die Eigenschaften von kleinen Programmen formal und informell beweisen.			
Inhalte: Dieses Modul richtet sich an Studentinnen und Studenten, deren Einstufungstest geringe Vorkenntnisse in einer objektorientierten Programmiersprache nachweist. Im Einführungskurs werden diese Kenntnisse konzentriert vermittelt und grundlegende Kompetenzen der Programmierung erworben. Es werden folgende Themen behandelt: <ul style="list-style-type: none"> • Maschinelle Modellierung des Berechenbarkeitsbegriffs (Registermaschinen) und zentrale imperative Programmierkonzepte. • Nachweis der Eigenschaften von kleinen Programmen (Hoare-Kalkül und/oder wp-Kalkül). • Objektorientierte Programmierkonzepte (wie Klassen, Objekte, Referenzen, Methoden, Vererbung, polymorphe Typsysteme, Abstrakte Klassen, Schnittstellen, generische Klassendefinitionen, Kapselung, Fehlerbehandlung usw.). • Einfache Datenstrukturen und deren Implementierung unter Verwendung objektorientierter Programmierkonzepte sowie grundlegende Konzepte der Datenabstraktion. • Vertiefte objektorientierte Modellierungstechniken und grundlegende Entwurfsmuster (Iteratoren, Beobachtermuster, Strukturmuster, MVC usw.). • Die Realisierung/Umsetzung der Konzepte werden anhand von modernen, gegenwärtig verwendeten, objektorientierten Programmiersprachen vorgestellt. 			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Einführungskurs	1	Einstufungstest, Anfertigung und Vorstellen der Lösungen von mindestens 85 % der Aufgaben	Einführungskurs 15
Vorlesung	3		Präsenzzeit V 45
Seminar am PC	2		Vor- und Nachbereitung V 30
			Präsenzzeit SPC 30
			Vor- und Nachbereitung SPC 90
			Prüfungsvorbereitung und Prüfung 30
Modulprüfung:		Klausur (120 Minuten); die Klausur kann auch in Form einer elektronischen Prüfungsleistung (120 Minuten) durchgeführt werden.	
Veranstaltungssprache:		Deutsch	
Pflicht zur regelmäßigen Teilnahme:		Vorlesung: Teilnahme wird empfohlen, Seminar am PC: Ja	
Arbeitszeitaufwand insgesamt:		240 Stunden	8 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Jedes Sommersemester	
Verwendbarkeit:		Bachelorstudiengang Informatik, Bachelorstudiengang Informatik für das Lehramt	

²² Studien- und Prüfungsordnung für den Bachelorstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Datenbanksysteme²³

Modul: Datenbanksysteme				
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik				
Modulverantwortliche/r: Dozentin oder Dozent des Moduls				
Zugangsvoraussetzungen: Keine				
Qualifikationsziele:				
<p>Die Studentinnen und Studenten können am Ende des Moduls den Aufbau einer Datenbank erläutern, die Verarbeitung von Befehlen an die Datenbank erklären, Datenbankmodelle (wie z. B. ER-Modelle) aus Anwendungsbeschreibungen erstellen, ein relationales Modell aus einem Datenbankschema ableiten, eine Datenbank auf Grundlage eines relationalen Modells erstellen, eine Anfrage in relationaler Algebra formalisieren, Abfragen zur Datenanalyse erstellen und auf einer Datenbank ausführen, Anfragen zur Datenbank und Schemamanipulation erstellen und auf der Datenbank ausführen. Sie können die Motivation der Normalisierung von Daten erklären und können Algorithmen zum Normalisieren von Daten anwenden. Sie können alternative Speicherstrukturen erklären und diese semantisch vergleichen. Sie können Anwendungen mit Zugriff auf eine Datenbank implementieren. Sie können Methoden zum Beschleunigen von Datenbankabfragen schematisch darstellen, erklären und mit geeigneten Datenstrukturen implementieren. Sie können prinzipielle Methoden zur Transaktionsverwaltung auf Datenbanken erklären und anwenden und können die Prinzipien des gleichzeitigen Zugriffs auf Datenbanken anwenden. Sie können Methoden zur Datenwiederherstellung erklären und implementieren. Sie können grundlegende Methoden des Data Minings verstehen und anwenden. Sie können Trends im Bereich Datenbanksysteme verstehen, erläutern und anwenden.</p>				
Inhalte:				
<p>Datenbankentwurf mit Entity-Relationship-Modellen und der UML; theoretische Grundlagen relationaler Datenbanksysteme, relationale Algebra; funktionale Abhängigkeiten, Normalformen, relationale Datenbankentwicklung: Datendefinition, Fremdschlüssel, andere Integritätsbedingungen, objektrelationale Abbildung, Sicherheits- und Schutzkonzepte; Transaktionsbegriff, transaktionale Garantien, Synchronisierung des Mehrbenutzerbetriebs, Fehlertoleranzeigenschaften.</p>				
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)	
Vorlesung	3	Schriftliche Bearbeitung der Übungsblätter	Präsenzzeit V	45
			Vor- und Nachbereitung V	45
Übung	2	Mündliche Präsentation der Lösungen von Übungsaufgaben in den Übungen	Präsenzzeit Ü	15
			Betreut im PC-Labor	15
			Vor- und Nachbereitung Ü	60
			Prüfungsvorbereitung und Prüfung	30
Modulprüfung:		Mündliche Prüfung (ca. 20 Minuten) oder Klausur (90 Minuten); die Klausur kann auch in Form einer elektronischen Prüfungsleistung (90 Minuten) durchgeführt werden.		
Veranstaltungssprache:		Deutsch		
Pflicht zur regelmäßigen Teilnahme:		Vorlesung: Teilnahme wird empfohlen, Übung: Ja		
Arbeitszeitaufwand insgesamt:		210 Stunden	7 LP	
Dauer des Moduls:		Ein Semester		
Häufigkeit des Angebots:		Jedes Sommersemester		
Verwendbarkeit:		Bachelorstudiengang Informatik, Bachelorstudiengang Informatik für das Lehramt, Bachelorstudiengang Bioinformatik		

²³ Studien- und Prüfungsordnung für den Bachelorstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Grundlagen der theoretischen Informatik²⁴

Modul: Grundlagen der Theoretischen Informatik			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentin oder Dozent des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten verstehen bei erfolgreichem Abschluss des Moduls die Grundlagen der Beschreibung und syntaktischen Analyse von Programmiersprachen. Sie können formale Sprachen innerhalb der Chomsky-Hierarchie einordnen. Sie beherrschen die gängigen Verfahren, um formale Sprachen von einer Beschreibungsform in eine andere zu überführen, sowie Beschreibungen in Normalformen oder minimale Formen zu übersetzen. Aus einer Beschreibung können sie die gemeinte Sprache ableiten. Sie verstehen, dass unterschiedliche Beschreibungsformen von Berechnungsmodellen gleichartig sind und verstehen die Verfahren, um eine Form in die andere zu überführen. Sie verstehen die prinzipiellen Möglichkeiten und Grenzen der Berechenbarkeit. Insbesondere verstehen sie das Halteproblem und seine Unlösbarkeit.			
Inhalte: Theoretische Rechenmodelle: Automaten, Turing-Maschinen. Formale Sprachen, Sprachakzeptoren, reguläre Ausdrücke, Grammatiken, Chomsky-Hierarchie, Berechenbarkeit, Komplexität			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	3	Schriftliche Bearbeitung der Übungsblätter	Präsenzzeit V 45 Vor- und Nachbereitung V 30
Übung	2	Mündliche Präsentation der Lösungen von Übungsaufgaben in den Übungen	Präsenzzeit Ü 30 Vor- und Nachbereitung Ü 75 Prüfungsvorbereitung und Prüfung 30
Modulprüfung:		Klausur (90 Minuten); die Klausur kann auch in Form einer elektronischen Prüfungsleistung (90 Minuten) durchgeführt werden.	
Veranstaltungssprache:		Deutsch	
Pflicht zur regelmäßigen Teilnahme:		Vorlesung: Teilnahme wird empfohlen, Übung: Ja	
Arbeitszeitaufwand insgesamt:		210 Stunden	7 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Jedes Sommersemester	
Verwendbarkeit:		Bachelorstudiengang Informatik, Bachelorstudiengang Informatik für das Lehramt, Bachelorstudiengang Bioinformatik	

²⁴ Studien- und Prüfungsordnung für den Bachelorstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Softwaretechnik²⁵

Modul: Softwaretechnik			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentin oder Dozent des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten verstehen die wesentlichen Fragestellungen für die Entwicklung großer Systeme; verstehen die wesentlichen unterschiedlichen Randbedingungen, unter denen diese Entwicklung erfolgen kann; verstehen die wichtigsten Ansätze, mit denen diese Fragestellungen gelöst werden, und können ihre Eigenschaften analysieren; können beurteilen, unter welchen Umständen welche Ansätze Erfolg versprechend sind; können die wichtigsten dieser Ansätze selbst durchführen; beherrschen die Methoden des Projektmanagements und können Gender- und Diversityaspekte im Projektmanagement berücksichtigen.			
Inhalte: In der Vorlesung werden Prinzipien, Methoden und Techniken für die Entwicklung großer Programmsysteme einschließlich einer Anleitung zum Projektmanagement vermittelt. Wichtige Einzelfertigkeiten werden in der begleitenden Übung konkret erprobt. Die Teilnehmenden lernen Antworten u. a. auf folgende Fragen: – Wie findet man heraus, welche Eigenschaften eine Software haben soll? (Anforderungsermittlung) – Wie beschreibt man dann diese Eigenschaften? (Anforderungsbeschreibung) – Was macht gute Software aus? (Qualitätsmerkmale) – Wie strukturiert man die Software so, dass sie sich leicht bauen und flexibel verändern lässt? (Architektur, Entwurf) – Wie deckt man Mängel in Software auf? (Analytische Qualitätssicherung) – Wie beugt man Mängeln vor? (Konstruktive Qualitätssicherung) – Wie organisiert man die Arbeit einer Softwareabteilung oder eines Softwareprojekts, um regelmäßig kostengünstige und hochwertige Resultate zu erzielen? (Projektmanagement, Prozessmanagement, Organisation) Den Studentinnen und Studenten wird empfohlen, das Modul „Softwaretechnik“ und das Softwareprojekt/Berufspraktikum in demselben Semester zu absolvieren.			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	4	Beteiligung an den Diskussionen in der Übung, Präsentation eigener Rechercheergebnisse	Präsenzzeit V 60
Übung	2		Vor- und Nachbereitung V 60 Präsenzzeit Ü 30 Vor- und Nachbereitung Ü 90 Prüfungsvorbereitung und Prüfung 60
Modulprüfung:		Mündliche Prüfung (ca. 20 Minuten) oder Klausur (90 Minuten); die Klausur kann auch in Form einer elektronischen Prüfungsleistung (90 Minuten) durchgeführt werden.	
Veranstaltungssprache:		Deutsch	
Pflicht zur regelmäßigen Teilnahme:		Vorlesung: Teilnahme wird empfohlen, Übung: Ja	
Arbeitszeitaufwand insgesamt:		300 Stunden	10 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Jedes Sommersemester	
Verwendbarkeit:		Bachelorstudiengang Informatik, Bachelorstudiengang Informatik für das Lehramt	

²⁵ Studien- und Prüfungsordnung für den Bachelorstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Auswirkungen der Informatik²⁶

Modul: Auswirkungen der Informatik			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentin oder Dozent des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten <ul style="list-style-type: none"> – verstehen den Unterschied zwischen Verfügungswissen und Orientierungswissen, – lernen, beim Nachdenken über Informatiksysteme zu unterscheiden zwischen technischen Fragestellungen, Technikfolgenabschätzung und Technikfolgenbewertung, – verstehen die Verantwortungsaspekte der Ingenieur Tätigkeit, – erlernen einige Aspekte der Technikfolgenabschätzung in bestimmten Informatik-Themenbereichen wie z. B. Sicherheit, Schutz der Privatsphäre, – verstehen Gender- und Diversityaspekte von Anwendungen und in der Anwendungsentwicklung. 			
Inhalte: Dieses Modul behandelt die Auswirkungen der Informatik. Nach grundlegenden Fragen (Konzept, Verfügungswissen, Verantwortungsbegriff, Subjektivität von Techniksoziologie) werden konkret an Beispielen Technikfolgen in informatiklastigen Gebieten behandelt, z. B. die Sicherheit softwareintensiver technischer Systeme, der Schutz der Privatsphäre oder Auswirkungen der Computerisierung der Arbeitswelt.			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	2	Beteiligung an den Diskussionen im Seminar, Präsentation eigener Rechercheergebnisse	Präsenzzeit V 30
Seminar	2		Vor- und Nachbereitung V 45
			Präsenzzeit S 30
			Vor- und Nachbereitung S 45
Modulprüfung:		Keine	
Veranstaltungssprache:		Deutsch	
Pflicht zur regelmäßigen Teilnahme:		Vorlesung: Teilnahme wird empfohlen, Übung: Ja	
Arbeitszeitaufwand insgesamt:		150 Stunden	5 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Jedes Wintersemester	
Verwendbarkeit:		Bachelorstudiengang Informatik, Bachelorstudiengang Informatik für das Lehramt	

²⁶ Studien- und Prüfungsordnung für den Bachelorstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Gesellschaftliche Aspekte der Informatik²⁷

Modul: Gesellschaftliche Aspekte der Informatik			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentin oder Dozent des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten können exemplarisch informatische Systeme und deren Auswirkungen auf die Gesellschaft verstehen und einordnen. Umgekehrt verstehen sie, wie der Bedarf der Gesellschaft informatische Systeme prägt. Sie können beschreiben und beurteilen, wie Informatik und informatische Systeme sowohl Benachteiligungen beseitigen können als auch diese erst erzeugen können. Sie sind in der Lage, solche Interaktionen zwischen den Systemen, den Individuen und der Gesellschaft zu beschreiben, einzuordnen und daraus konkrete Handlungsstrategien abzuleiten. Sie können ihre Rechercheergebnisse adressatenorientiert präsentieren.			
Inhalte: Fortschritte der Informatik und neue Anwendungen der Informatik sowie gesellschaftliche Aspekte und Auswirkungen ihrer Anwendung (z. B. Mensch-Maschine-Interaktion, Barrierefreiheit, Chancengleichheit, Wirkung von informatischen Systemen auf das Geschlechterverhältnis, Personalführung, Dokumentation)			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Seminaristische Übung	3	Beteiligung an den Diskussionen im Seminar, Präsentation eigener Rechercheergebnisse, Referate	Präsenzzeit 45 Vor- und Nachbereitung 75 Prüfungsvorbereitung und Prüfung 30
Modulprüfung:		Schriftliche Ausarbeitung eines Referats (ca. 2 000 Wörter); die Modulprüfung wird nicht differenziert bewertet.	
Veranstaltungssprache:		Deutsch	
Pflicht zu regelmäßiger Teilnahme:		Ja	
Arbeitszeitaufwand insgesamt:		150 Stunden	5 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Unregelmäßig	
Verwendbarkeit:		Bachelorstudiengang Informatik: Studienbereich ABV (Fachnahe Zusatzqualifikationen), Bachelorstudiengang Mathematik: Studienbereich ABV (Fachnahe Zusatzqualifikationen); Masterstudiengang Informatik: Praktische Informatik	

²⁷ Studien- und Prüfungsordnung für den Bachelorstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Softwareprojekt A²⁸

Modul: Softwareprojekt A									
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik									
Modulverantwortliche/r: Dozentin oder Dozent des Moduls									
Zugangsvoraussetzungen: Erfolgreiche Absolvierung des Moduls Softwaretechnik									
Qualifikationsziele: Die Studentinnen und Studenten können ihre softwaretechnischen Kenntnisse und Qualifikationen erfolgreich in einem kleinen Softwareprojekt einsetzen und die entsprechenden Verfahren anwenden. Sie können unter Anleitung ein größeres Projekt in Aufgaben und Teilprodukte zerlegen und Schnittstellen zwischen diesen definieren. Sie verstehen die Notwendigkeit dieser Schnittstellen und der möglichen Probleme bei der später folgenden Integration. Sie verstehen Qualitäts-, Aufwands-, Akzeptanz- und Erfolgsfaktoren eines Softwareprojekts. Sie können mündlich und schriftlich in einem Projektteam mit mehr als fünf Personen kommunizieren, sich koordinieren und ein Softwareprojekt unter Anleitung erfolgreich planen. Sie können ihre Arbeiten selbst verwalten. Sie sind fähig, die Qualität ihrer Lösungsbeiträge im Kontext des Gesamtprojekts zu beurteilen. Sie können die Werkzeuge zur Projektdurchführung und zur Projektverwaltung benutzen. Sie können ihre Ergebnisse mündlich und schriftlich geeignet darstellen.									
Inhalte: Im Softwareprojekt wird das implizite Wissen (tacit knowing) der Softwareentwicklung angeeignet. Dazu wird von den Studentinnen und Studenten im Team unter Anleitung der Dozentin oder des Dozenten ein größeres Softwaresystem arbeitsteilig entwickelt. Dabei sollen alle Phasen eines Softwareprojekts, so wie sie in einem Unternehmen heute stattfinden, durchlaufen sowie typische Methoden, wie sie im Modul Softwaretechnik kennen gelernt wurden, anhand der im Unternehmensinsatz typischen Werkzeuge und Hilfsmittel durchgeführt werden. Es werden exemplarisch unternehmenstypische Softwarekomponenten und Werkzeuge zur Durchführung aller auftretenden Aufgaben vorgestellt und erprobt.									
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)						
Projektseminar	2	Erfolgreiches Bearbeiten und Dokumentieren der Teilaufgaben, Zwischenpräsentationen, Teilnahme an den Diskussionen. Präsentation der Ergebnisse und der Projekterfahrung mit nachfolgender Diskussion	<table border="0"> <tr> <td>Präsenzzeit</td> <td>30</td> </tr> <tr> <td>Vor- und Nachbereitung</td> <td>240</td> </tr> <tr> <td>Prüfungsvorbereitung und Prüfung</td> <td>30</td> </tr> </table>	Präsenzzeit	30	Vor- und Nachbereitung	240	Prüfungsvorbereitung und Prüfung	30
Präsenzzeit	30								
Vor- und Nachbereitung	240								
Prüfungsvorbereitung und Prüfung	30								
Modulprüfung:		Keine							
Veranstaltungssprache:		Deutsch							
Pflicht zu regelmäßiger Teilnahme:		Ja							
Arbeitszeitaufwand insgesamt:		300 Stunden	10 LP						
Dauer des Moduls:		8 Wochen im Block bzw. ein Semester							
Häufigkeit des Angebots:		Jedes Semester							
Verwendbarkeit:		Bachelorstudiengang Informatik: Studienbereich ABV (Fachnahe Zusatzqualifikationen), Bachelorstudiengang Mathematik: Studienbereich ABV (Fachnahe Zusatzqualifikationen)							

²⁸ Studien- und Prüfungsordnung für den Bachelorstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Softwareprojekt B²⁹

Modul: Softwareprojekt B									
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik									
Modulverantwortliche/r: Dozentin oder Dozent des Moduls									
Zugangsvoraussetzungen: Erfolgreiche Absolvierung des Moduls Softwaretechnik									
Qualifikationsziele:									
<p>Die Studentinnen und Studenten können ihre softwaretechnischen Kenntnisse und Qualifikationen erfolgreich in einem kleinen Softwareprojekt einsetzen und die entsprechenden Verfahren anwenden. Sie können unter Anleitung ein größeres Projekt in Aufgaben und Teilprodukte zerlegen und Schnittstellen zwischen diesen definieren. Sie verstehen die Notwendigkeit dieser Schnittstellen und der möglichen Probleme bei der später folgenden Integration. Sie verstehen Qualitäts-, Aufwands-, Akzeptanz- und Erfolgsfaktoren eines Softwareprojekts. Sie können mündlich und schriftlich in einem Projektteam mit mehr als fünf Personen kommunizieren, sich koordinieren und ein Softwareprojekt unter Anleitung erfolgreich planen. Sie können ihre Arbeiten selbst verwalten. Sie sind fähig, die Qualität ihrer Lösungsbeiträge im Kontext des Gesamtprojekts zu beurteilen. Sie können die Werkzeuge zur Projektdurchführung und zur Projektverwaltung benutzen. Sie können ihre Ergebnisse mündlich und schriftlich geeignet darstellen.</p>									
Inhalte:									
<p>Im Softwareprojekt wird das implizite Wissen (tacit knowing) der Softwareentwicklung angeeignet. Dazu wird von den Studentinnen und Studenten im Team unter Anleitung der Dozentin oder des Dozenten ein größeres Softwaresystem arbeitsteilig entwickelt. Dabei sollen alle Phasen eines Softwareprojekts, so wie sie in einem Unternehmen heute stattfinden, durchlaufen sowie typische Methoden, wie sie im Modul Softwaretechnik kennen gelernt wurden, anhand der im Unternehmensinsatz typischen Werkzeuge und Hilfsmittel durchgeführt werden. Es werden exemplarisch unternehmenstypische Softwarekomponenten und Werkzeuge zur Durchführung aller auftretenden Aufgaben vorgestellt und erprobt.</p>									
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)						
Projektseminar	2	Erfolgreiches Bearbeiten und Dokumentieren der Teilaufgaben, Zwischenpräsentationen, Teilnahme an den Diskussionen	<table border="0"> <tr> <td>Präsenzzeit</td> <td>30</td> </tr> <tr> <td>Vor- und Nachbereitung</td> <td>240</td> </tr> <tr> <td>Prüfungsvorbereitung und Prüfung</td> <td>30</td> </tr> </table>	Präsenzzeit	30	Vor- und Nachbereitung	240	Prüfungsvorbereitung und Prüfung	30
Präsenzzeit	30								
Vor- und Nachbereitung	240								
Prüfungsvorbereitung und Prüfung	30								
Modulprüfung:		Präsentation der Ergebnisse und der Projekterfahrung (15 Minuten); die Modulprüfung wird differenziert bewertet.							
Veranstaltungssprache:		Deutsch							
Pflicht zu regelmäßiger Teilnahme:		Ja							
Arbeitszeitaufwand insgesamt:		300 Stunden	10 LP						
Dauer des Moduls:		8 Wochen im Block bzw. ein Semester							
Häufigkeit des Angebots:		Jedes Semester							
Verwendbarkeit:		Bachelorstudiengang Informatik: Studienbereich ABV (Fachnahe Zusatzqualifikationen), Bachelorstudiengang Mathematik: Studienbereich ABV (Fachnahe Zusatzqualifikationen)							

²⁹ Studien- und Prüfungsordnung für den Bachelorstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Rechnerarchitektur³⁰

Modul: Rechnerarchitektur			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentin oder Dozent des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten verstehen am Ende des Moduls: die grundlegenden Architekturmerkmale von Rechnersystemen, die Interaktionen der Architekturmerkmale in Mehrkern- und Mehrprozessorsystemen, die elementaren Möglichkeiten der Beschleunigung von Rechnersystemen.			
Inhalte: Themenbereiche sind hier insbesondere Harvard/v. Neumann-Architektur, Mikroarchitektur RISC/CISC, Mikroprogrammierung, Pipelining, Cache, Speicherhierarchie, Bussysteme, Assemblerprogrammierung, Multiprozessorsysteme, VLIW, Sprungvorhersage. Ebenso werden interne Zahlendarstellungen, Rechnerarithmetik und die Repräsentation weiterer Datentypen im Rechner behandelt.			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	2	Schriftliche Bearbeitung der Übungsblätter, mündliche Präsentation der Lösungen von Übungsaufgaben in den Übungen.	Präsenzzeit V 30
Seminar am PC	2		Vor- und Nachbereitung V 30
			Präsenzzeit SPC 15
			Betreutes Selbststudium am PC 15
			Prüfungsvorbereitung und Prüfung 60
Modulprüfung:		Klausur (60 Minuten); die Klausur kann auch in Form einer elektronischen Prüfungsleistung (60 Minuten) durchgeführt werden	
Veranstaltungssprache:		Deutsch	
Pflicht zur regelmäßigen Teilnahme:		Vorlesung: Teilnahme wird empfohlen, Seminar am PC: Ja	
Arbeitszeitaufwand insgesamt:		150 Stunden	5 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Jedes Wintersemester	
Verwendbarkeit:		Bachelorstudiengang Informatik für das Lehramt 60-LP-Modulangebot Informatik	

³⁰ Studien- und Prüfungsordnung für den Bachelorstudiengang Informatik für das Lehramt des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Betriebs- und Kommunikationssysteme³¹

Modul: Betriebs- und Kommunikationssysteme			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentin oder Dozent des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Die Studentinnen und Studenten verstehen am Ende des Moduls: die Rolle des Betriebssystems als Abstraktion des Rechnersystems, den grundlegenden Aufbau aktueller Betriebssysteme, die Funktion und den Aufbau des Internets. Sie können Rechner auf Assembler-Ebene und systemnah programmieren, können die Vor- und Nachteile verschiedener Mechanismen (PIO vs. DMA, polling vs. Interrupt, paging vs. Segmentation usw.) beurteilen, Mechanismen von Betriebssystemen sinnvoll einsetzen, können Programme über das Netzwerk kommunizieren lassen. Ein-/Ausgabe-Systeme, DMA/PIO, Unterbrechungsbehandlung, Puffer, Prozesse/Threads, virtueller Speicher, UNIX und Windows, Shells, Utilities, Peripherie und Vernetzung, Netze, Medien, Medienzugriff, Protokolle, Referenzmodelle, TCP/IP, grundlegender Aufbau des Internets			
Inhalte: Themenbereiche sind hier insbesondere Ein-/Ausgabe-Systeme, DMA/PIO, Unterbrechungsbehandlung, Puffer, Prozesse/Threads, virtueller Speicher, UNIX und Windows, Shells, Utilities, Peripherie und Vernetzung, Netze, Medien, Medienzugriff, Protokolle, Referenzmodelle, TCP/IP, grundlegender Aufbau des Internets.			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	2	Schriftliche Bearbeitung der Übungsblätter, mündliche Präsentation der Lösungen von Übungsaufgaben in den Übungen.	Präsenzzeit V 60
Seminar am PC	2		Vor- und Nachbereitung V 30 Präsenzzeit Ü 30 Vor- und Nachbereitung Ü 120 Prüfungsvorbereitung und Prüfung 30
Modulprüfung:		Klausur (60 Minuten); die Klausur kann auch in Form einer elektronischen Prüfungsleistung (60 Minuten) durchgeführt werden	
Veranstaltungssprache:		Deutsch	
Pflicht zur regelmäßigen Teilnahme:		Vorlesung: Teilnahme wird empfohlen, Seminar am PC: Ja	
Arbeitszeitaufwand insgesamt:		180 Stunden	5 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Jedes Sommersemester	
Verwendbarkeit:		Bachelorstudiengang Informatik für das Lehramt 60-LP-Modulangebot Informatik	

³¹ Studien- und Prüfungsordnung für den Bachelorstudiengang Informatik für das Lehramt des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Grundlagen der technischen Informatik³²

Modul: Grundlagen der Technischen Informatik			
Hochschule/Fachbereich/Institut: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentin oder Dozent des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele:			
Die Studentinnen und Studenten können am Ende des Moduls:			
<ul style="list-style-type: none"> – logische Funktionen auf physikalische Schaltkreise abbilden, – einfache Schaltungen verstehen und berechnen, – den Einsatz der Halbleitertechnik in Schaltungen nachvollziehen, – den Übergang von der analogen zur digitalen Welt und umgekehrt beschreiben, – Analoge Schaltungen aufbauen und analysieren, – In Assembler und C hardwarenah programmieren, – Eingebettete Systeme in Betrieb nehmen und – Software auf eingebetteten Systemen installieren. 			
Inhalte:			
Das Modul Grundlagen der Technischen Informatik bildet die Basis für das Verständnis der Funktionsweise realer Rechnersysteme. Ausgehend von der Logik werden in diesem Modul vorrangig die Themenbereiche Schaltnetze und Schaltwerke, Logikminimierung, Gatter, Flip-Flops, Speicher, Automaten und einfacher Hardware-Entwurf behandelt. Weiterhin werden grundlegende Kenntnisse aus den Bereichen Gleich- und Wechselstromnetzwerke, Halbleiter, Transistoren, CMOS, Operationsverstärker, A/D- und D/A-Umsetzer vermittelt, soweit sie für die Informatik notwendig sind. Im dazugehörigen Praktikum werden mit zahlreichen praktischen Versuchen das in der Vorlesung sowie das im Modul Rechnerarchitektur, Betriebs- und Kommunikationssysteme Erlernte vertieft. Aufbauend auf einer einfachen Hardwareplattform mit Prozessor und diversen Schnittstellen sollen die Teilnehmerinnen und Teilnehmer lernen, elementare Treiber zu programmieren, Betriebssystemroutinen zu erweitern und die Schnittstellen anzusteuern. Anschließend sollen die Systeme vernetzt werden und mit ihrer Umwelt in Interaktion treten können.			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	2	Erklärung und Absolvierung von mindestens 85 % der Versuche	Präsenzzeit V 30
			Vor- und Nachbereitung V 60
Seminar am PC	4		Präsenzzeit SPC 60
			Vor- und Nachbereitung SPC 120
			Prüfungsvorbereitung und Prüfung 30
Modulprüfung:		Praktische Prüfung (Darstellung theoretischer Hintergründe, Versuchsergebnis und Protokollbuch)	
Veranstaltungssprache:		Deutsch	
Pflicht zur regelmäßigen Teilnahme:		Vorlesung: Teilnahme wird empfohlen, Seminar am PC: Ja	
Arbeitszeitaufwand insgesamt:		300 Stunden	10 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Jedes Wintersemester	
Verwendbarkeit:		Bachelorstudiengang Informatik	

³² Studien- und Prüfungsordnung für den Bachelorstudiengang Informatik des Fachbereichs Mathematik und Informatik der Freien Universität Berlin



Modul: Algorithmen, Datenstrukturen und Datenabstraktion A³³

Modul: Algorithmen, Datenstrukturen und Datenabstraktion A			
Hochschule/Fachbereich/Lehreinheit: Freie Universität Berlin/Mathematik und Informatik/Informatik			
Modulverantwortliche/r: Dozentin oder Dozent des Moduls			
Zugangsvoraussetzungen: Keine			
Qualifikationsziele: Studentinnen und Studenten können die Grundbegriffe der Algorithmik definieren. Sie wissen, was ein abstrakter Datentyp ist, und verstehen den Unterschied zwischen Spezifikation und Implementierung. Sie kennen die wichtigsten abstrakten Datentypen und die Datenstrukturen zu deren Implementierung und können diese in Bezug auf ihre Eigenschaften beurteilen und geeignet auswählen und einsetzen. Sie können die Korrektheit von Algorithmen nachweisen und die asymptotische Laufzeit von Algorithmen bestimmen. Sie kennen die Definition und verstehen die praktische Bedeutung von NP-Vollständigkeit für die effiziente Lösbarkeit von Problemen.			
Inhalte: Die grundlegenden Datenstrukturen Listen, Schlangen, Keller, Bäume; Sortierverfahren (Mergesort, Quicksort, u. a.), Suchverfahren, Auswahlverfahren; Abstrakte Datentypen Prioritätswarteschlange und Wörterbuch und zugehörige Datenstrukturen wie Heaps, Hashtabellen, binäre Suchbäume, B-Bäume u. a.; Algorithmen auf Graphen wie Breiten- und Tiefensuche, topologisches Sortieren, kürzeste Spannbäume, kürzeste Wege; Algorithmen für Zeichenketten; Speicherverwaltung; Allgemeine Lösungsstrategien wie Teile und Herrsche, dynamische Programmierung, Auswählen und Abschneiden, gierige Algorithmen. Mathematische Analyse von Algorithmen bezüglich Laufzeit und Speicherplatz. NP-Vollständigkeit.			
Lehr- und Lernformen	Präsenzstudium (Semesterwochenstunden = SWS)	Formen aktiver Teilnahme	Arbeitsaufwand (Stunden)
Vorlesung	4	Schriftliche Bearbeitung der Übungsblätter, zwei mündliche Präsentationen der Lösung jeweils einer Übungsaufgabe in der Übung	Präsenzzeit V 60 Vor- und Nachbereitung V 45 Präsenzzeit Ü 30 Vor- und Nachbereitung Ü 135
Übung	2		Prüfungsvorbereitung und Prüfung 30
Modulprüfung:		Klausur (120 Minuten); die Klausur kann auch in Form einer elektronischen Prüfungsleistung (120 Minuten) durchgeführt werden	
Veranstaltungssprache:		Deutsch	
Pflicht zur regelmäßigen Teilnahme:		Vorlesung: Teilnahme wird empfohlen, Übung: Ja	
Arbeitsaufwand insgesamt:		300 Stunden	10 LP
Dauer des Moduls:		Ein Semester	
Häufigkeit des Angebots:		Jedes Wintersemester	
Verwendbarkeit:		Masterstudiengang für das Lehramt an Integrierten Sekundarschulen und Gymnasien mit dem Profil Quereinstieg	

³³ Studien- und Prüfungsordnung der Freien Universität Berlin für den Masterstudiengang für das Lehramt an Integrierten Sekundarschulen und Gymnasien mit dem Profil Quereinstieg