



## Modulhandbuch

---

*Informatik (SPO WS 18/19)*

*Bachelor*

---

*Fakultät Informatik*

Stand: 2023-09-26

# Inhalt

<b>1</b>	<b>Übersicht .....</b>	<b>4</b>
<b>2</b>	<b>Einführung .....</b>	<b>5</b>
2.1	Zielsetzung .....	5
2.2	Zulassungsvoraussetzungen.....	6
2.3	Zielgruppe .....	7
2.4	Studienaufbau .....	8
2.4.1	Erster Studienabschnitt.....	9
2.4.2	Zweiter Studienabschnitt .....	10
2.4.3	Vorrückungsvoraussetzungen .....	12
2.5	Fachwissenschaftliche Wahlpflichtmodule.....	13
2.5.1	Fachwissenschaftliche Wahlpflichtmodule im Angebot der THI .....	13
2.5.2	Fachwissenschaftliche Wahlpflichtmodule im Angebot der virtuellen Hochschule Bayern .....	13
2.6	Praktisches Studiensemester .....	14
2.7	Duales Studium .....	15
2.8	Konzeption .....	16
<b>3</b>	<b>Qualifikationsprofil .....</b>	<b>17</b>
3.1	Leitbild.....	18
3.2	Studienziele .....	19
3.2.1	Fachspezifische Kompetenzen des Studiengangs .....	19
3.2.2	Fachübergreifende Kompetenzen des Studiengangs .....	19
3.2.3	Prüfungskonzept des Studiengangs .....	21
3.2.4	Anwendungsbezug des Studiengangs .....	23
3.2.5	Beitrag einzelner Module zu den Studiengangzielen .....	24
3.3	Mögliche Berufsfelder.....	26
<b>4</b>	<b>Modulbeschreibungen .....</b>	<b>27</b>
4.1	Allgemeine Pflichtmodule .....	27
	Einführungsprojekt .....	27
	Grundlagen der Programmierung 1.....	29
	Rechnerarchitektur .....	32
	Mathematische Grundlagen 1 .....	34
	Physikalische und Elektrotechnische Grundlagen .....	36
	Englisch .....	38
	Grundlagen der Programmierung 2.....	40
	Mathematische Grundlagen 2 .....	42
	Betriebssysteme .....	44
	Technik der Mensch-Maschine Interaktion .....	46
	Rechnernetze .....	48
	Algorithmen und Datenstrukturen .....	51
	Software Engineering.....	53
	Webtechnologien .....	55
	Verteilte Systeme.....	57
	Wissensbasierte Systeme .....	59

---

Datenbanksysteme .....	61
Fachwissenschaftliches Seminar .....	63
Grundlagen der Theoretischen Informatik .....	65
Diskrete Mathematik .....	67
Computergrafik .....	69
Projektmanagement .....	71
Vorbereitendes Praxisseminar (PLV1) .....	73
Praktikum .....	75
Nachbereitendes Praxisseminar (PLV 2) .....	77
Informations- und Medienkompetenz (PLV3) .....	79
Programmierparadigmen und deren Sprachen .....	81
Statistik und Kombinatorik .....	83
Projekt .....	85
IT-Recht .....	87
IT-Sicherheit .....	89
Bachelorarbeit .....	91
Seminar Bachelorarbeit .....	93
Grundlagen der Betriebswirtschaft und des Gründertums .....	95
4.2 Fachwissenschaftliche WPM .....	97
Eingebettete Systeme und Echtzeitsysteme .....	97
Parallel Programming .....	99
Principles of Modern Software Development .....	101

## 1 Übersicht

<b>Name des Studiengangs</b>	Informatik (Bachelor)
<b>Studienart &amp; Abschlussgrad</b>	Grundständig, B.Sc. (Bachelor of Science), Vollzeit
<b>Erstmaliges Startdatum</b>	01.10.2006 als Bachelor-Studiengang, jährlicher Start
<b>Regelstudienzeit</b>	7 Semester, 210 ECTS
<b>Lage des Praxissemesters</b>	5. Semester
<b>Studienort</b>	THI, Ingolstadt
<b>Unterrichtssprache/n</b>	Deutsch, einzelne Module können in englischer Sprache angeboten werden
<b>Kooperation</b>	Keine; duales Studium ist möglich
<b>Zulassungsvoraussetzungen</b>	Hochschulzugangsberechtigung
<b>Kapazität</b>	56 Studierende pro Studienjahr
<b>Studiengangleiter</b>	Prof. Dr. Franz Regensburger E-Mail: <a href="mailto:franz.regensburger@thi.de">franz.regensburger@thi.de</a> Phone: +49 (0) 841 / 9348-2780
<b>Studienfachberater</b>	Prof. Dr. Franz Regensburger E-Mail: <a href="mailto:franz.regensburger@thi.de">franz.regensburger@thi.de</a> Phone: +49 (0) 841 / 9348-2780
<b>Praktikumsbeauftragter</b>	Prof. Dr. Bernd Hafenrichter E-Mail: <a href="mailto:bernd.hafenrichter@thi.de">bernd.hafenrichter@thi.de</a> Phone: +49 (0) 841 / 9348-2522

## 2 Einführung

Das vorliegende Modulhandbuch beschreibt das Lehrangebot im Bachelor-Studiengang Informatik It. Studien- und Prüfungsordnung für den Bachelorstudiengang Informatik an der Technischen Hochschule Ingolstadt vom 25.07.2011 in der Fassung der Änderungssatzung vom 20.11.2017.

Insbesondere nennt es die Studienziele und Studieninhalte der einzelnen Pflichtmodule, der fachwissenschaftlichen Wahlpflichtmodule und der praxisbegleitenden Lehrveranstaltungen des Studiengangs sowie die zeitliche Aufteilung der Semesterwochenstunden je Fach und Studiensemester.

Bei Mehrdeutigkeiten hat die übergeordnete Studien- und Prüfungsordnung Vorrang.

### 2.1 Zielsetzung

Der Bachelor-Studiengang Informatik hat das Ziel, durch praxis- und anwendungsorientierte Lehre eine auf der Grundlage wissenschaftlicher Erkenntnisse und Methoden beruhende Ausbildung zu vermitteln, die zu einer eigenverantwortlichen Berufstätigkeit als Informatiker befähigt. Durch eine umfassende Ausbildung in Grundlagenfächern sollen die Studierenden in die Lage versetzt werden, komplexe Anwendungsfelder und Bedürfnisse der Anwender von informationsverarbeitenden Systemen zu analysieren, solche Systeme zu entwerfen, zu implementieren, zu beschaffen, in eine Systemumgebung zu integrieren und zu betreuen.

Ebenso soll der Bachelor-Studiengang Informatik eine solide Basis für ein späteres konsekutives Master-Studium und eine eventuell daran anschließende Promotion legen. Aus diesem Grund wird im Studiengang Informatik darauf geachtet, dass neben einer Grundausbildung in anwendungsbezogenen mathematischen Techniken auch solide Kompetenzen auf den Gebieten der Diskreten Mathematik, der Theoretischen Informatik und der angewandten Statistik vermittelt werden.

Darüber hinaus sollen die Studierenden jene Flexibilität erlangen, die benötigt wird, um der rasch fortschreitenden informationsverarbeitenden Entwicklung gerecht zu werden. So sollen sie nach ihrem Studium auch in der Lage sein, das Management auf verschiedenen informationstechnischen Gebieten zu unterstützen und nach entsprechender Einarbeitung selbst Führungsaufgaben oder freiberufliche Aufgaben zu übernehmen.

Das Studium soll deswegen neben dem gezielten Erwerb von Fachwissen auch wichtige Schlüsselqualifikationen sowie soziale und methodische Kompetenzen für die berufliche Praxis vermitteln, die Persönlichkeitsbildung fördern, in Führungswissen und -techniken einführen und dazu befähigen, die Auswirkungen der Informatik auf die Umwelt und die Gesellschaft zu erkennen und entsprechend zu einem verantwortungsbewussten Handeln im Berufsfeld Informatik erziehen.

## 2.2 Zulassungsvoraussetzungen

Zur Zulassung für den Bachelorstudiengang Informatik müssen die allgemeinen Zulassungsvoraussetzungen für ein Studium an Hochschulen für angewandte Wissenschaften erfüllt sein, d.h., entweder die allgemeine oder fachgebundene Hochschulreife oder die Fachhochschulreife.

Die verbindlichen Regelungen für diesen Studienplan sind zu finden in:

- Der Studien- und Prüfungsordnung (SPO) für den Bachelorstudiengang Informatik in der konsolidierten Fassung vom 20.11.2017 inkl. Anlage zur konsolidierten Fassung vom 20.11.2017 (<https://www.thi.de/hochschule/ueber-uns/hochschulorganisation/stabsstelle-recht/satzungen-fakultaet-informatik/bachelorstudiengaenge-i/spo-informatik-bachelor/>)
- Der Allgemeine Prüfungsordnung (APO) der Technischen Hochschule Ingolstadt, <https://www.thi.de/hochschule/ueber-uns/hochschulorganisation/stabsstelle-recht/allgemeine-satzungen/> (in der letztgültigen Fassung)
- Sowie der Rahmenprüfungsordnung (RaPO) für die Fachhochschulen, <https://www.thi.de/hochschule/ueber-uns/hochschulorganisation/stabsstelle-recht/allgemeine-satzungen/> (in der letztgültigen Fassung)

## 2.3 Zielgruppe

Der Studiengang Informatik soll vor allem Studierende ansprechen, die

1. ein Interesse an den Grundlagen und Methoden der Allgemeinen Informatik (auch Kerninformatik genannt) in ihrer gesamten Bandbreite haben, die von den mittlerweile klassischen Säulen der Kerninformatik (Technische Informatik, Praktische Informatik, Theoretische Informatik) aufgespannt wird
2. verstehen möchten, wie grundlegende Konzepte und Technologien der Elektrotechnik, insbesondere der Digitaltechnik, die Basis für moderne Computer bilden
3. verstehen möchten, wie auf Basis der Digitaltechnik Rechnerarchitekturen konzipiert und mittels CISC- und RISC-Befehlssätzen zu programmierbaren Prozessoren werden
4. die Aufgaben und die Leistungsmerkmale moderner Betriebssysteme studieren möchten, welche die Basis sowohl für kleine eingebettete Systeme (SMART-Watch, Wearables, DSL-Router), PC-artige Endgeräte (Tablets, Notebooks, Workstations) als auch Server-Systeme (Mail-Server, Web-Server, Datenbank-Management-Systeme, Application Server) bis hin zu Cluster-Systemen für Highperformance Computing (HPC) bilden
5. die Aufgaben und die Leistungsmerkmale von Netzkomponenten (Zugangspunkten, Switches, Router, Firewalls) verstehen möchten, die zusammen die Infrastruktur des modernen Internets bilden
6. die wesentlichen Konzepte von Programmiersprachen und den ihnen zugrundeliegenden Programmierparadigmen studieren möchten und dabei für jedes Paradigma mindestens eine konkrete Programmiersprache erlernen und beherrschen möchten
7. bereitwillig solide Grundkenntnisse aus den Bereichen der Mathematik und Logik erwerben möchten, auf deren Grundlage die Methoden und Techniken der effizienten Datenhaltung in Datenstrukturen und Datenbanken sowie die Formulierung effizienter und gleichzeitig korrekter Algorithmen basieren
8. in mindestens fünf verschiedenen Pflichtvorlesungen, teilweise mit assoziierten Praktika, moderne und erfolgreiche Programmiersprachen erlernen möchten, die die gesamte Bandbreite der heute industriell eingesetzten Programmiersprachen abdecken (prozedurale, objektorientierte, logische und funktionale Programmierung) und die sowohl für die Erstellung von Werkzeugen der Informatik (Betriebssysteme, Compiler, Entwicklungsumgebungen) als auch für die Implementierung von Programmsystemen der angewandten Informatik (Datenbankmanagementsysteme, Web-Server, Cloud-Infrastrukturen, KI-Systeme) unverzichtbar sind
9. die Kompetenz erwerben möchten, aus einem durch die Module des Studiums vermittelten Repertoire aus algorithmischen Fähigkeiten, Programmierkompetenzen, Methoden und Techniken anwendungsbezogen auszuwählen und damit konkrete Problemstellungen im Rahmen von Projekten zu lösen und effiziente, bedarfsgerechte Anwendungen zu implementieren

## 2.4 Studienaufbau

Die Regelstudienzeit für die Bachelor-Studiengänge umfasst sieben Semester. Die Studiengänge gliedern sich in zwei Studienabschnitte.

Das folgende Schaubild bildet den Studienverlauf grafisch ab:

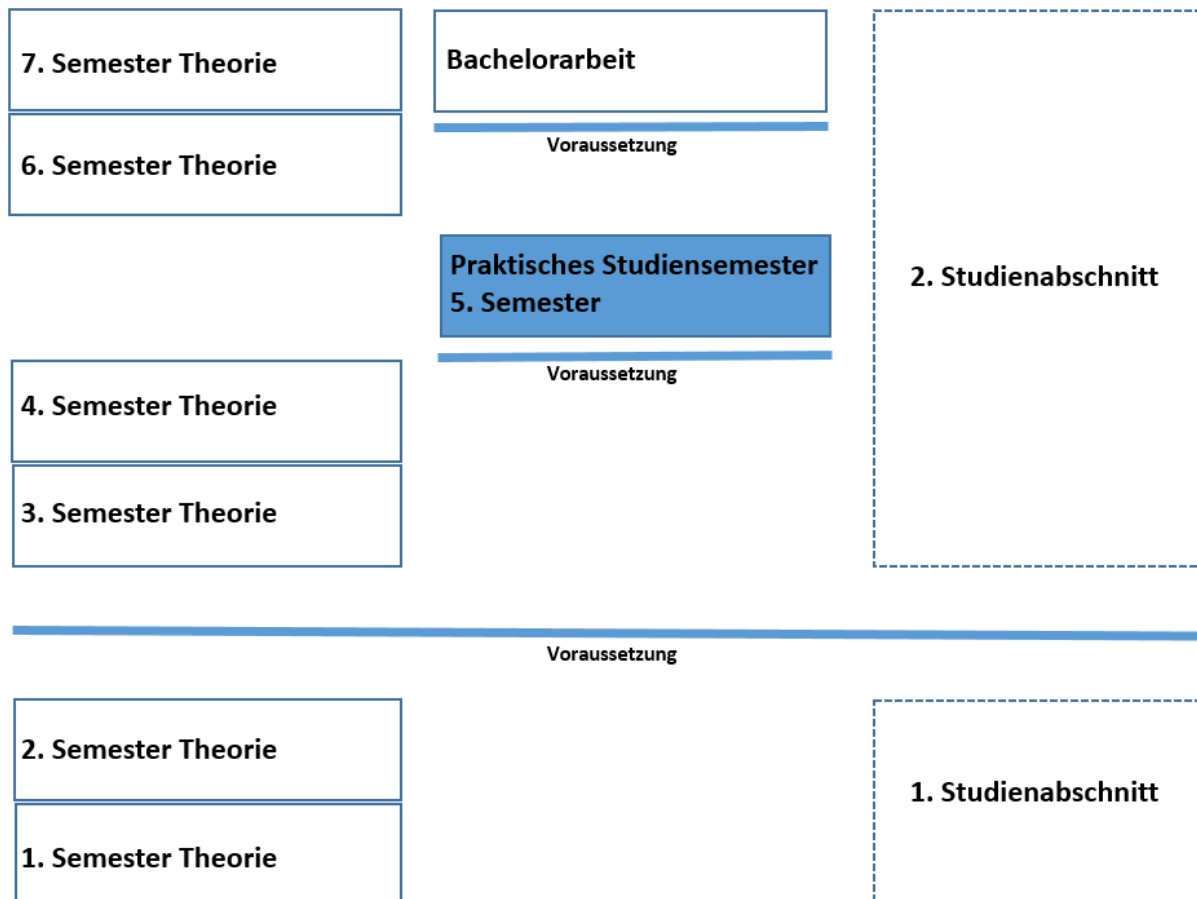


Abbildung 1 Aufbau des Studiums

Der erste Studienabschnitt umfasst zwei theoretische Studiensemester. Der zweite Studienabschnitt beinhaltet vier theoretische Semester und ein praktisches Semester, welches in der Regel als 5. Studiensemester geführt wird.

Bei Erfüllung bestimmter Zugangsvoraussetzungen besteht die Möglichkeit, im Anschluss an das Bachelor-Studium Informatik ein Master-Studium anzuschließen.



### 2.4.1 Erster Studienabschnitt

Der erste Studienabschnitt umfasst zwei theoretische Semester und beginnt im Wintersemester.

Modul	Nr.	Fach	Aufteilung nach Semestern			
			1. Sem	2. Sem	SWS	CP
Einführungsprojekt	1	Einführungsprojekt	LN		2	2
Grundlagen der Programmierung 1	2.1	Grundlagen der Programmierung 1	P		4	5
	2.2	Praktikum Grundlagen der Programmierung 1	LN		2	2
Rechnerarchitektur	3.1	Rechnerarchitektur	P		4	5
	3.2	Praktikum Rechnerarchitektur	LN		1	2
Mathematische Grundlagen 1	4.1	Mathematische Grundlagen 1	P		4	5
	4.2	Übung zu Mathematische Grundlagen 1			1	1
Physikalische und Elektrotechnische Grundlagen	5	Physikalische und Elektrotechnische Grundlagen	P		4	5
Englisch	6	Englisch	P		2	3
Grundlagen der Programmierung 2	7.1	Grundlagen der Programmierung 2		P	4	5
	7.2	Praktikum Grundlagen der Programmierung 2		LN	2	2
Mathematische Grundlage 2	8.1	Mathematischen Grundlagen 2		P	4	5
	8.2	Übungen zu Mathematische Grundlagen 2			1	1
Betriebssysteme	9	Betriebssysteme		P	4	5
Technik der Mensch-Maschine Interaktion	10	Technik der Mensch-Maschine Interaktion		P	4	5
Rechnernetze	11.1	Rechnernetze		P	4	5
	11.2	Praktikum Rechnernetze		LN	2	2
<b>Summe</b>					<b>49</b>	<b>60</b>

Legende:

- SWS Semesterwochenstunden
- CP Leistungspunkte nach European Credit Transfer System (ECTS)
- P schriftliche Prüfung
- LN studienbegleitender Leistungsnachweis

## 2.4.2 Zweiter Studienabschnitt

Der zweite Studienabschnitt beginnt mit dem dritten Semester und umfasst 4 theoretische Semester und ein Praxissemester.

### Semester 3-5

Modul	Nr.	Fach	Aufteilung nach Semestern				
			3. Sem	4. Sem	5. Sem	SWS	CP
Algorithmen und Datenstrukturen	12	Algorithmen und Datenstrukturen	P			4	5
Software Engineering	13.1	Software Engineering	P			4	5
	13.2	Praktikum Software Engineering	LN			2	2
Webtechnologien	14.1	Webtechnologien	P			4	5
	14.2	Praktikum Webtechnologien	LN			2	2
Verteilte Systeme	15.1	Verteilte Systeme	P			4	5
	15.2	Praktikum Verteilte Systeme	LN			1	1
Wissensbasierte Systeme	16	Wissensbasierte Systeme	P			4	5
Datenbanksysteme	17.1	Datenbanksysteme		P		4	5
	17.2	Praktikum Datenbanksysteme		LN		2	2
Fachwissenschaftliches Seminar	18	Fachwissenschaftliches Seminar		SA		2	3
Grundlagen der Theoretischen Informatik	19	Grundlagen der Theoretischen Informatik		P		4	5
Diskrete Mathematik	20	Diskrete Mathematik		P		4	5
Computergrafik	21.1	Computergrafik		P		2	5
	21.2	Praktikum Computergrafik		LN		2	
Projektmanagement	22	Projektmanagement		P		4	5
Praktisches Studiensemester	23	Vorbereitendes Praxisseminar (PLV 1)			LN	1	2
	24	Praktikum			B		24
	25	Nachbereitendes Praxisseminar (PLV 2)			LN	1	2
	26	Informations- und Medienkompetenz (PLV 3)			LN	1	2
<b>Summe</b>			<b>25</b>	<b>24</b>	<b>3</b>	<b>52</b>	<b>90</b>

Legende:

- SWS Semesterwochenstunden
- CP Leistungspunkte nach European Credit Transfer System (ECTS)
- P schriftliche Prüfung
- SA Seminararbeit
- LN studienbegleitender Leistungsnachweis
- B Bericht

**Semester 6-7**

Modul	Nr.	Fach	Aufteilung nach Semestern			
			6. Sem	7. Sem	SWS	CP
Programmierparadigmen und deren Sprachen	27	Programmierparadigmen und deren Sprachen		P	4	5
Statistik und Kombinatorik	28	Statistik und Kombinatorik	P		4	5
Projekt	29	Projekt	Proj		4	5
IT-Recht	30	IT-Recht	P		4	5
Fachwissenschaftliche Wahlpflichtfächer	31.1	Fachwissenschaftliches Wahlpflichtfach 1	LN		4	5
	31.2	Fachwissenschaftliches Wahlpflichtfach 2	LN		4	5
	31.3	Fachwissenschaftliches Wahlpflichtfach 3		LN	4	5
IT-Sicherheit	32	IT-Sicherheit	P		4	5
Bachelorarbeit	33	Bachelorarbeit		BA		12
Seminar Bachelorarbeit	34	Seminar Bachelorarbeit		LN	2	3
Grundlagen der Betriebswirtschaft und des Gründertums	35	Grundlagen der Betriebswirtschaft und des Gründertums	LN		4	5
<b>Summe</b>			<b>28</b>	<b>10</b>	<b>38</b>	<b>60</b>

Legende:

SWS	Semesterwochenstunden
CP	Leistungspunkte nach European Credit Transfer System (ECTS)
P	schriftliche Prüfung
PrA	praktische Arbeit
SA	Seminararbeit
LN	studienbegleitender Leistungsnachweis <sup>1</sup>
BA	Bachelorarbeit
Proj	Projekt

**Für Studien- und Prüfungsleistungen aus sämtlichen Semestern, die in mehreren Teilen oder in Fächern mit begleitenden Praktika zu erbringen sind, gelten ggf. Voraussetzungen, die in der Anlage zur SPO oder in den folgenden Modulbeschreibungen geregelt sind.**

<sup>1</sup> Die konkrete Prüfungsform wird in der Anlage SPO bzw. im aktuellen Studienplan spezifiziert.

### 2.4.3 Vorrückungsvoraussetzungen

Um sicherzustellen, dass die für das Verständnis der einzelnen Studienabschnitte erforderlichen Kenntnisse vorhanden sind, gibt es mehrere Vorrückungsvoraussetzungen. Bei Nichterfüllen dieser Voraussetzungen entsteht meist eine Verzögerung im Studienfortschritt, die zum Füllen der jeweiligen Lücken genutzt werden soll. Um die Gesamtdauer des Studiums im Rahmen zu halten, sind zusätzlich einige Fristen zu beachten. Einen Überblick über diese Voraussetzungen und Fristen gibt die nachfolgende Aufstellung<sup>2</sup>:

1. Zum Eintritt in den zweiten Studienabschnitt ist nur berechtigt, wer mindestens 42 Leistungspunkte aus Modulen des ersten Studienabschnitts erzielt hat.
2. Zum Eintritt in das Praktikum im praktische Studiensemester ist nur berechtigt, wer alle Prüfungen des ersten Studienabschnitts bestanden und mindestens 20 Leistungspunkte aus Modulen der ersten beiden Semester des zweiten Studienabschnitts erzielt hat.
3. Voraussetzung für die Ausgabe des Themas der Bachelorarbeit ist die erfolgreiche Ableistung des praktischen Studiensemesters.

Die verbindlichen Regelungen sind im Wortlaut zu finden in der Studien- und Prüfungsordnung (SPO-18) Informatik, in der Rahmenprüfungsordnung (RaPO), in der Allgemeinen Prüfungsordnung (APO) der Technischen Hochschule Ingolstadt und in der Immatrikulationssatzung der THI.

<https://www.thi.de/hochschule/ueber-uns/hochschulorganisation/stabsstelle-recht>

---

<sup>2</sup> rechtlich verbindlich für Vorrückungs- und Zulassungsvoraussetzungen ist nur die SPO-18

## 2.5 Fachwissenschaftliche Wahlpflichtmodule

### 2.5.1 Fachwissenschaftliche Wahlpflichtmodule im Angebot der THI

Im 6. und 7. Semester sind regulär fachwissenschaftliche Wahlpflichtmodule (FW-Module) zu belegen.

**Mindestens ein Fachwissenschaftliches Wahlpflichtmodul mit Unterrichtssprache Englisch muss belegt werden (siehe Anlage SPO).**

Am Ende des vorausgehenden Semesters erfolgt die Einschreibung für die FW-Module (online über Moodle), um die Teilnehmerzahl zu ermitteln. Die einzelnen FW-Module können nur bei ausreichender Teilnehmerzahl angeboten werden.

Das Angebot an FW-Modulen wird für jedes Semester neu erstellt, je nach Verfügbarkeit der Dozenten bzw. Lehrbeauftragten aus der Industrie.

Bei Interesse können nach Rücksprache mit dem Studiengangleiter auch geeignete Fächer anderer Studiengänge als FW-Fächer gewählt werden. Ein Anspruch darauf besteht nicht.

**Melden Sie sich dazu bitte in den ersten beiden Wochen des Semesters beim Studiengangleiter.**

### 2.5.2 Fachwissenschaftliche Wahlpflichtmodule im Angebot der virtuellen Hochschule Bayern

Das Angebot der Wahlpflichtmodule kann selbstständig um fachwissenschaftliche Wahlpflichtfächer der VHB (Virtuelle Hochschule Bayern) ergänzt werden. Dafür gilt folgendes:

- Studierende informieren sich selbstständig über das VHB Angebot unter [www.vhb.org](http://www.vhb.org).
- Vor Belegung des Fachs muss sich der Studierende bis spätestens 3 Wochen nach Semesterbeginn beim Studiengangleiter erkundigen, ob das VHB-Fach als fachwissenschaftliches Wahlpflichtfach des Studiengangs grundsätzlich angerechnet werden kann.
- Nach erfolgreicher Absolvierung des VHB-Fachs ist ein Antrag auf Anrechnung zu stellen.
- VHB-Fächer erscheinen nicht im Prüfungsangebot der Fakultät. Eine Anmeldung über die Systeme der THI ist nicht möglich.
- Prüfungstermin und Prüfungsort werden vom VHB-Kursleiter bestimmt. Eine terminliche Überschneidungsfreiheit mit THI-Prüfungen wird nicht garantiert.
- Studierende entscheiden selbstständig, ob sie sich ein VHB-Fach als fachwissenschaftliches Wahlpflichtfach anrechnen lassen wollen.

## 2.6 Praktisches Studiensemester

Das Praxissemester ist während des Studiums für alle Studierenden zu durchlaufen. Es wird in Unternehmen aus Industrie, Mittelstand und öffentlicher Verwaltung durchgeführt.

Das praktische Studiensemester des zweiten Studienabschnitts umfasst einen Zeitraum von 20 Wochen und wird durch drei Lehrveranstaltungen an der Hochschule begleitet, von denen eine vor (Vorbereitendes Praxisseminar - PLV1) und zwei nach der Praxisphase (Nachbereitendes Praxisseminar - PLV2, Informations- und Medienkompetenz - PLV3) stattfinden.

Begleitend zum Praxissemester ist ein Praktikumsbericht anzufertigen. Die Anforderungen an den Praktikumsbericht sind in der Anlage zur SPO aufgeführt.

## 2.7 Duales Studium

In Kooperation mit ausgewählten Praxispartnern kann der Studiengang Informatik auch im dualen Studienmodell („Studium mit vertiefter Praxis“) absolviert werden. Dual Studierende arbeiten während der vorlesungsfreien Zeit im Kooperationsunternehmen und können so ihr im Studium erworbenes theoretisches Wissen mit Berufspraxis ergänzen. Zusätzlich wird das Praxissemester sowie die Abschlussarbeit im Unternehmen absolviert. Eine optimale Verzahnung von Theorie und Praxis ist gewährleistet durch die Qualitätsstandards von „hochschule dual“, der Dachmarke des dualen Studiums in Bayern (<https://www.hochschule-dual.de/>).

Die Vorlesungszeiten im dualen Studienmodell entsprechen den normalen Studien- und Vorlesungszeiten an der THI. Das Curriculum des dualen Studiengangmodells unterscheidet sich gegenüber dem regulären Studiengangkonzept in folgenden Punkten:

- **Praxissemester im Kooperationsunternehmen:** Dual Studierende absolvieren das Praxissemester im Kooperationsunternehmen
- **Dual-Module:** Regelmäßig angeboten werden gesonderte FW-Fächer für Dual-Studierende. Diese Veranstaltungen werden an der Hochschule bzw. einem Dualpartner durchgeführt. Angeboten werden auch gesonderte Projekte sowie separate Praxisseminare für Dualstudierende. Eine Anrechnung von Projekten und Praxisseminaren über außer-hochschulisch erworbene Kompetenzen aus dem Lernort Unternehmen ist möglich. Einzelne Veranstaltungen werden nach Möglichkeit von Lehrbeauftragten der Kooperationsunternehmen durchgeführt
- **Abschlussarbeit im Kooperationsunternehmen:** Im dualen Studienmodell wird die Abschlussarbeit bei dem Kooperationsunternehmen geschrieben, i.d.R. über ein praxisrelevantes Thema mit Bezug zum Studienschwerpunkt. Die Erstbetreuung erfolgt durch einen Dozenten aus dem Studiengang Informatik

Organisatorisch zeichnet sich das duale Studiengangmodell durch folgende Bestandteile aus:

- **Einführungsveranstaltung:** Im Rahmen der Semesteröffnung und der Informationsveranstaltungen des Studiengangleiters zu Studienbeginn wird eine gesonderte Veranstaltung für Dualstudierende angeboten
- **Mentoring:** Zentrale Ansprechpartner für Dualstudierende in der Fakultät sind die jeweiligen Studiengangleiter. Diese organisieren jährlich ein Mentoring-Treffen mit den Dualstudierenden des jeweiligen Studiengangs
- **Qualitätsmanagement:** In den Evaluationen und Befragungen an der THI zur Qualitätssicherung der Studiengänge sind separate Frageblöcke für das duale Studium enthalten
- **„Forum dual“:** Organisiert vom Career Service und Studienberatung (CSS) findet einmal jährlich das „Forum dual“ statt. Dieses fördert den fachlich-organisatorischen Austausch zwischen den dualen Kooperationspartnern und der Fakultät und dient zur Qualitätssicherung der dualen Studienprogramme. Zu dem Termin geladen sind alle Kooperationspartner im dualen Studium sowie Vertreter und Dualstudierende der Fakultät

Weiterführende Informationen zum Dualen Studium und den aktuellen Unternehmenspartnern des Studiengangs User Experience Design Bachelor sind unter <https://www.thi.de/studium/studienangebote/duales-studium> zu finden.

Formalrechtliche Regelungen zum dualen Studium für alle Studiengänge der THI sind in der APO (s. §§ 17, 29 und 30) und der Immatrikulationssatzung (s. §§ 8b, 9 und 18) geregelt.

## 2.8 Konzeption

Der Studiengang Informatik wird seit 1998 an der TH Ingolstadt angeboten. Das zunächst für einen Diplomstudiengang konzipierte Curriculum wurde über die Jahre mehrfach überarbeitet, insbesondere wurden Schwerpunkte für Fachdomänen wie die Flug- und Fahrzeugtechnik oder Wirtschaftsinformatik aufgenommen und in späteren Reformen in eigenständige Studiengänge ausgelagert.

Dieser Prozess wurde im Wesentlichen getrieben durch einen stetigen Kontakt mit Vertretern von in der Region angesiedelten Industrieunternehmen, der Beobachtung der sich stetig ändernden bzw. neu entstehenden Arbeitsprofile für Informatiker und einem Meinungsaustausch mit aktuellen und ehemaligen Studierenden (Alumni-Netzwerk der Technischen Hochschule Ingolstadt).

Das Curriculum des hier zur Re-Akkreditierung vorgesehenen Bachelor-Studiengangs Informatik wurde erstmalig im Jahr 2006 akkreditiert und letztmalig im Jahr 2017 im Rahmen einer Programmakkreditierung re-akkreditiert und soll nun ein weiteres Mal auf Basis der SPO vom 25.07.2011 in der Fassung der Änderungssatzung vom 20.11.2017 re-akkreditiert werden.

Aufgrund der Umstellung von einer Programmakkreditierung auf eine Systemakkreditierung im Jahr 2023 wurde das vorliegende Modulhandbuch nach den neuen Maßgaben der hauseigenen Akkreditierungsstelle überarbeitet. Die Studien- und Prüfungsordnung des Studiengangs bleibt jedoch unverändert in der Fassung der Änderungssatzung vom 20.11.2017 erhalten.

Der Bachelor-Studiengang Informatik ist auf 7 Semester mit einem Umfang von insgesamt 210 ECTS-Punkten angelegt, wobei jedes Semester mit 30 ECTS-Punkten kreditiert wird. Das Curriculum umfasst 35 Module, die aus 26 Vorlesungen, zwei Übungen, neun vorlesungsbegleitenden Praktika, zwei Seminaren, einem Praxissemester nebst drei praxisbegleitenden Lehrveranstaltungen (PLVs), zwei Projekten und der Bachelorarbeit bestehen.

Er vermittelt in einer ersten Stufe eine akademische Hochschulausbildung auf Bachelor-Niveau, die sowohl berufsqualifizierend ist als auch die Basis für ein konsekutives Master-Studium der Informatik darstellt.



### 3 Qualifikationsprofil

Der Bachelor-Studiengang Informatik wurde mit dem Ziel entwickelt, die Studierenden für eine eigenverantwortliche Berufstätigkeit in Berufsfeldern zu befähigen, in denen sowohl Kompetenzen der Kerninformatik als auch solide und praxisrelevante Kompetenzen aus dem Gebiet der Angewandten Informatik benötigt werden.

Dafür bieten folgende Teilbereiche des Curriculums die notwendigen Voraussetzungen:

- Vermittlung solider Kenntnisse der Mathematik und der Theoretischen Informatik, welche die Studierenden befähigen, die in der Kerninformatik und der Angewandten Informatik eingesetzten Verfahren zu verstehen und anzuwenden. Dies umfasst die Bereiche Lineare Algebra, Analysis, Diskrete Mathematik, Kombinatorik und Statistik sowie Automatentheorie, Formale Sprachen, Berechenbarkeit, Entscheidbarkeit und Grundlagen der Komplexitätstheorie
- Grundlagen der technischen Informatik wie elektrotechnische und physikalische Grundlagen, sowie Rechnerarchitektur, Rechnernetze und Technik der Mensch-Maschine Interaktion
- Praktischen Informatik mit umfangreicher Ausbildung auf dem Gebiet der Programmierung im Rahmen mehrerer Programmiervorlesungen und Praktika, Betriebssysteme, Verteilte Systeme, Wissensbasierte Systeme, Programmierparadigmen, Datenbanksysteme, Algorithmen und Datenstrukturen, Software-Engineering, Web-Technologien. Die Betonung der Ausbildung praktischer Fähigkeiten wird dadurch dokumentiert, dass von den genannten Modulen fast alle auch ein Praktikum als Teil des Moduls beinhalten
- Methoden und Verfahren der Angewandten Informatik im Rahmen der Veranstaltungen Computergrafik, Projektmanagement (mit Fokus auf den hierbei eingesetzten Werkzeugen), IT-Sicherheit und IT-Recht; Durchführung eines Projekts über den Zeitraum eines ganzen Semesters hinweg, bei dem die Studierenden auch die Rolle der Projektleitung übernehmen (meist in agiler Ausprägung unter Verwendung von Scrum oder Kanban)
- Betriebswirtschaftliche Kompetenzen im Rahmen der Veranstaltungen Grundlagen der Betriebswirtschaft und des Gründertums, Projektmanagement (mit Fokus auf betriebswirtschaftliche Aspekte), Projekt
- Übergreifende Fähigkeiten wie Sprachkompetenz, Sozialkompetenz, Präsentieren, Argumentieren sowie die Befähigung zum wissenschaftlichen Arbeiten werden in den Modulen Englisch, PLV1, PLV2, PLV3, Seminar, Seminar Bachelorarbeit und Bachelorarbeit adressiert.

### 3.1 Leitbild

Der Studiengang Informatik integriert das Leitbild der Lehre auf folgende Weise:

*Wir bereiten unsere Studierenden auf die Herausforderungen der Zukunft vor:*

1. Vermittlung der erforderlichen informatischen Fachkompetenz zur Entwicklung von Werkzeugen für Informatiker auf dem Gebiet der Kerninformatik (Treiber, Kernelmodule, Parser, Stored Procedures und User Defined Functions für DBMS, Web-basierte Werkzeuge mit REST API, Cloud-Komponenten)
2. Vermittlung der Kompetenz zur Entwicklung von komplexen Softwaresystemen für Client/Server Architekturen und Cloud-basierten Architekturen, unabhängig von der konkreten Anwendungsdomäne auf den Gebieten der Angewandten Informatik.
3. Eine Spezialisierung in Richtung einer speziellen Anwendungsdomäne kann individuell im Rahmen des Praxissemesters und der Bachelor-Arbeit erfolgen

*Wir befähigen unsere Studierenden, Problemlösungen auf der Basis wissenschaftlicher Erkenntnisse zu erarbeiten:*

- Vermittlung fundierter wissenschaftlicher Grundlagen auf den Gebieten der Kern-Informatik und der Angewandten Informatik
- Argumentationskompetenz und Problemlösungskompetenz über Übungen, Seminare und Praktika

*Wir eröffnen unseren Studierenden herausragende regionale und internationale Perspektiven:*

- Regional, überregional und international aktuell und künftig große Nachfrage nach IT-Fachkräften in den Bereichen der Kern-Informatik und der Angewandten Informatik

*Wir lehren und lernen im persönlichen Austausch:*

- Intensiver Austausch zwischen Lehrenden, Studierenden und Praxisexperten
- Projekt- und praxisbezogenes Arbeiten
- Kennenlernen der Facetten des projekthaften Arbeitens: Arbeiten alleine vs. das Arbeiten in unterschiedlichen Gruppengrößen

*Wir helfen allen Studierenden, ihr individuelles Potenzial zu entdecken und auszuschöpfen:*

- Spezialisierung über die individuelle Wahl von Wahlpflichtfächern, im Praxissemester sowie im Rahmen der Bachelorarbeit
- Förderung der Persönlichkeitsentwicklung. Unsere Lehrveranstaltungen vermitteln auch nicht-fachliche Kompetenzen, wie z.B. Methoden- und Teamkompetenz, Kommunikationskompetenz, Sozialkompetenz, insbesondere die Module Projekt und Praxissemester

## 3.2 Studienziele

### 3.2.1 Fachspezifische Kompetenzen des Studiengangs

Die Studieninhalte wurden entsprechend den Anforderungen aus Industrie- und Mittelstand sowie des Qualifikationsrahmens für deutsche Hochschulabschlüsse definiert.

Für den Bachelor-Studiengang Informatik müssen die allgemeinen Zulassungsvoraussetzungen für ein Studium an Hochschulen für angewandte Wissenschaften erfüllt sein.

Die Studierenden werden auf die Möglichkeiten der Sprachausbildung an der Technischen Hochschule Ingolstadt besonders hingewiesen.

Mit Abschluss des Studiums sind die Studierenden in der Lage,

- 1 vernetzte Software-Systeme sowohl im Kontext der Kerninformatik als auch in den anwendungsorientierten Kontexten der Angewandten Informatik zu konzipieren und zu entwickeln - auf Basis eines breiten technischen Grundverständnisses
- 2 Durch die Kenntnis von Entwicklungsmethoden für komplexe Systeme flexibel auf zukünftige Entwicklungen in allen Arbeitsfeldern der Software-Erstellung reagieren zu können
- 3 komplexe, zukünftige Probleme im Umfeld der Erstellung von Werkzeugen (Entwicklungsumgebungen, Software-Management, -Deployment und -Wartung) und Infrastrukturen (Treiber, Frameworks, Middleware) zu lösen
- 4 Projekte fachübergreifend zu planen, zu koordinieren und kostenbewusst durchzuführen

### 3.2.2 Fachübergreifende Kompetenzen des Studiengangs

*Methodenkompetenzen:*

Mit Abschluss des Studiums sind die Studierenden in der Lage,

- Methoden zum Design, Umsetzung, Integration und Testen von Software im Kontext der Kerninformatik sowie bei der Erstellung von Anwendungen im Kontext der Angewandten Informatik einzusetzen
- Problemstellungen zu analysieren, übergreifende Zusammenhänge zu erkennen, Grundlagen und Prinzipien bei der Problemlösung umzusetzen, Lösungen technisch und formgestalterisch zu bewerten sowie Entscheidungsvorlagen aufzubereiten
- wissenschaftlich zu arbeiten

*Sozialkompetenzen:*

Mit Abschluss des Studiums sind die Studierenden in der Lage,

- komplexe Aufgabenstellungen allein und im Team zu bearbeiten (Kommunikations- und Teamfähigkeit)
- technische Teams und Kreativteams zu leiten
- zu planen, zu organisieren, und Führung auszuüben
- wissenschaftlichen Diskurs zu führen

*Selbstkompetenzen:*

Mit Abschluss des Studiums sind die Studierenden in der Lage,

- Projekte zu strukturieren und zu steuern (Zeitmanagement)
- sich selbst zu organisieren (Selbstorganisation)
- zu kommunizieren und zu präsentieren (auch in englischer Sprache)
- analytisch und lösungsorientiert zu denken
- zielorientiert und selbstständig zu arbeiten
- kreativ zu arbeiten und Ideen zu entwickeln
- Entscheidungen zu treffen

### 3.2.3 Prüfungskonzept des Studiengangs

Im Studiengang bilden technische Fächer den Schwerpunkt. Soweit möglich sollen die Studierenden anhand von Projektarbeit lernen, das erworbene Wissen praktisch umzusetzen sowie ihre Ergebnisse zu präsentieren. Technische Fächer werden in üblicher Weise schriftlich geprüft. Daneben werden die Prüfungsformen Seminararbeit, Projektarbeit und praktische Prüfung eingesetzt. Bei der Entwicklung des Studiengangs wurde darauf geachtet, dass angemessene Prüfungsformen zum Einsatz kommen, um so die Prüfung der Studiengangziele sicherzustellen.

Im Rahmen der Fachwissenschaftlichen Wahlpflichtmodule können auch mündliche Prüfungen eingesetzt werden. Im Allgemeinen können die Studierenden hierbei aus einem sehr umfangreichen Angebot an Fächern wählen, welches sich aus dem Gesamtangebot aller Informatik-artigen Studiengänge der Fakultät Informatik speist. Besondere Bedeutung liegt auf dem anwendungsbasierten Lernen, d.h. die Vermittlung theoretischer Bestandteile wird durch Praktika und Übungen vertieft.

Die genaue Zuweisung der Prüfungsformen zu den Modulen und die spezifische Ausprägung der Prüfungsform ist der Anlage zur SPO zu entnehmen.

Modul	Prüfungsform
1 Einführungsprojekt	Leistungsnachweis (mit/ohne Erfolg)
2 Grundlagen der Programmierung 1	Schriftliche Prüfung, 90min.
3 Rechnerarchitektur	Schriftliche Prüfung, 90min.
4 Mathematische Grundlagen 1	Schriftliche Prüfung, 90min.
5 Physikalische und elektrotechnische Grundlagen	Schriftliche Prüfung, 90min.
6 Englisch	Schriftliche Prüfung, 90min.
7 Grundlagen der Programmierung 2	Schriftliche Prüfung, 90min.
8 Mathematische Grundlagen 2	Schriftliche Prüfung, 90min.
9 Betriebssysteme	Schriftliche Prüfung, 90min.
10 Technik der Mensch-Maschine Interaktion	Schriftliche Prüfung, 90min.
11 Rechnernetze	Schriftliche Prüfung, 90min.
12 Algorithmen und Datenstrukturen	Schriftliche Prüfung, 90min.
13 Software Engineering	Schriftliche Prüfung, 90min.
14 Webtechnologien	Schriftliche Prüfung, 90min.
15 Verteilte Systeme	Schriftliche Prüfung, 90min.

16 Wissensbasierte Systeme	Schriftliche Prüfung, 90min.
17 Datenbanksysteme	Schriftliche Prüfung, 90min.
18 Fachwissenschaftliches Seminar	Seminararbeit
19 Grundlagen der Theoretischen Informatik	Schriftliche Prüfung, 90min.
20 Diskrete Mathematik	Schriftliche Prüfung, 90min.
21 Computergrafik	Schriftliche Prüfung, 90min.
22 Projektmanagement	Schriftliche Prüfung, 90min.
23 Vorbereitendes Praxisseminar (PLV 1)	Leistungsnachweis (mit/ohne Erfolg)
24 Praktikum	Praktikumsbericht
25 Nachbereitendes Praxisseminar (PLV 2)	Leistungsnachweis (mit/ohne Erfolg)
26 Informations- und Medienkompetenz (PLV 3)	Leistungsnachweis (mit/ohne Erfolg)
27 Programmierparadigmen und deren Sprachen	Schriftliche Prüfung, 90min.
28 Statistik und Kombinatorik	Schriftliche Prüfung, 90min.
29 Projekt	Projektarbeit
30 IT-Recht	Schriftliche Prüfung, 90min.
31.1 Fachwissenschaftliches Wahlpflichtmodul 1	Leistungsnachweis (benotet)
31.2 Fachwissenschaftliches Wahlpflichtmodul 2	Leistungsnachweis (benotet)
31.3 Fachwissenschaftliches Wahlpflichtmodul 3	Leistungsnachweis (benotet)
32 IT-Sicherheit	Schriftliche Prüfung, 90min.
33 Bachelorarbeit	Schriftliche Abschlussarbeit
34 Seminar Bachelorarbeit	Leistungsnachweis (mit/ohne Erfolg)
35 Grundlagen der Betriebswirtschaft und des Gründertums	Leistungsnachweis (benotet)

### 3.2.4 Anwendungsbezug des Studiengangs

Der Studiengang vermittelt praxisorientiert die benötigten Methoden und Techniken zur Entwicklung von Software-basierten Systemen aus dem Gebiet der Kerninformatik und der Angewandten Informatik.

Zu diesem Zweck beinhalten viele Module des Studiengangs neben einer seminaristisch angelegten Vorlesung auch ein Praktikum, in welchem die erlernten Methoden und Kompetenzen mittels geeigneter Tools und Programmen auf konkrete Fragestellungen praktisch angewandt werden. Darüber hinaus tragen das 20-wöchige Praxissemester, das Projekt sowie die Möglichkeit, die Bachelorarbeit zu einem Thema aus dem Umfeld eines Unternehmens zu schreiben, dazu bei, die für einen nahtlosen Übergang zu eigenverantwortlicher Tätigkeit erforderlichen Praxiskenntnisse zu erlangen.

Der Studiengang wurde u.a. auf Basis einer Reihe von Gesprächen mit Unternehmensvertretern entwickelt, um die Studienganggestaltung auf die Anforderungen der Unternehmenspraxis auszurichten. Die Beschäftigungsfähigkeit der Absolventinnen und Absolventen wurde von Seite der involvierten Unternehmensvertreter sehr positiv beurteilt.

### 3.2.5 Beitrag einzelner Module zu den Studiengangzielen

Der Beitrag der einzelnen Module zu den Studiengangzielen kann der folgenden Kompetenzmatrix entnommen werden.

	Fach- kompetenzen	Methoden- kompetenzen	Sozial- kompetenzen	Selbst- kompetenzen
1 Einführungsprojekt	○	○	+	+
2 Grundlagen der Programmierung 1	++	+	○	○
3 Rechnerarchitektur	++	+	○	○
4 Mathematische Grundlagen 1	++	+	○	○
5 Physikalische und elektro-technische Grundlagen	++	+	○	○
6 Englisch	++	+	+	+
7 Grundlagen der Programmierung 2	++	+	○	○
8 Mathematische Grundlagen 2	++	+	○	○
9 Betriebssysteme	++	+	○	○
10 Technik der Mensch-Maschine Interaktion	++	+	○	○
11 Rechnernetze	++	+	○	○
12 Algorithmen und Datenstrukturen	++	+	○	○
13 Software-Engineering	++	+	+	○
14 Webtechnologien	++	+	+	○
15 Verteilte Systeme	++	+	○	○
16 Wissensbasierte-Systeme	++	+	○	○
17 Datenbanksysteme	++	+	○	○



18 Fachwissenschaftliches Seminar	+	++	+	++
19 Grundlagen der Theoretischen Informatik	++	+	○	○
20 Diskrete Mathematik	++	+	○	○
21 Computergrafik	++	+	○	○
22 Projektmanagement	++	++	○	○
23 Vorbereitendes Praxisseminar	+	○	++	++
24 Praktikum	+	+	+	+
25 Nachbereitendes Praxisseminar	+	+	+	++
26 Informations- und Medienkompetenz	++	++	+	○
27 Programmierparadigmen und deren Sprachen	++	++	○	○
28 Statistik und Kombinatorik	++	+	○	○
29 Projekt	++	++	++	++
30 IT-Recht	+	+	+	○
31.1 Fachwissenschaftliches Wahlpflichtmodul 1	++	+	○	○
31.2 Fachwissenschaftliches Wahlpflichtmodul 2	++	+	○	○
31.3 Fachwissenschaftliches Wahlpflichtmodul 3	++	+	○	○
32 IT-Sicherheit	++	+	+	○
33 Bachelorarbeit	++	++	○	++
34 Seminar Bachelorarbeit	+	+	+	++
35 Grundlagen der Betriebswirtschaft und des Gründertums	+	+	○	+

### 3.3 Mögliche Berufsfelder

Die Absolventinnen und Absolventen des Bachelor-Studiengangs Informatik sind branchenübergreifend für Fach- und Führungsaufgaben in Berufsfeldern befähigt, in denen sowohl Werkzeuge und Anwendungen der Kerninformatik als auch anwendungsorientierte Software-Lösungen aus den Gebieten der Angewandten Informatik entwickelt und zum Einsatz gebracht werden.

Zum Beispiel in den folgenden Einsatzgebieten:

- Entwicklung grundlegender Infrastruktur wie Treiber und Betriebssysteme
- Entwicklungsumgebungen (Editoren, Compiler, Debugger)
- Entwicklung und Erweiterung von Frameworks und Middleware
- Entwicklung, Erweiterung und Integration von Datenbanksystemen (Multi DB-Ansätze)
- Plattformen auf Basis von Web-Technologien (Cloud-basierte Systeme, Micro-Services)
- Entwicklung von Systemarchitekturen
- Software-Design und Software-Architektur
- Testen von Software

## 4 Modulbeschreibungen

### 4.1 Allgemeine Pflichtmodule

<b>Einführungsprojekt</b>			
<b>Modulkürzel:</b>	IB_EINF	<b>SPO-Nr.:</b>	1
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	1
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Wintersemester
<b>Modulverantwortliche(r):</b>	Regensburger, Franz		
<b>Leistungspunkte / SWS:</b>	2 ECTS / 2 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	24 h	
	Selbststudium:	26 h	
	Gesamtaufwand:	50 h	
<b>Lehrveranstaltungen des Moduls:</b>	Einführungsprojekt		
<b>Lehrformen des Moduls:</b>	Prj - Projekt		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
LN - ohne/mit Erfolg teilgenommen			
Weitere Erläuterungen: Für das Einführungsprojekt herrscht an allen drei Tagen Anwesenheitspflicht. Des Weiteren müssen die Teilnehmer an der Veranstaltung zur Bibliothekseinführung teilnehmen und den dazu gehörenden Online-Test bestehen.			
<b>Voraussetzungen gemäß SPO:</b>			
Keine			
<b>Empfohlene Voraussetzungen:</b>			
Grundkenntnisse der deutschen Sprache, Fähigkeit zur sozialen Interaktion			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch des Moduls sind die Studierenden in der Lage			
<ul style="list-style-type: none"> <li>sich auf dem Campus zurecht zu finden und wichtige Anlaufstellen (Mensa, Bibliothek, Dekanat, Vorlesungsräume) zu benennen und aufzusuchen</li> <li>ihre Kommilitonen über mindestens einen Kommunikationskanal zu erreichen (Telefon, Messenger, E-Mail)</li> <li>per Mail Kontakt mit Studiengangleiter, Studienfachbetreuer und Dekanat aufzunehmen</li> </ul>			

<ul style="list-style-type: none"><li>• kleine Programme sowie Audio-Clips in der Programmierumgebung Scratch zu erstellen und auszuführen</li><li>• das Virtualisierungssystem VirtualBox bei Bedarf selbständig auf ihren eigenen Rechnern neu zu installieren</li><li>• die im Rahmen des Einführungsprojekts auf ihren Rechnern installierte virtuelle Maschine zu starten, Konfigurationsdateien zu erstellen sowie Programmdateien mit einem Texteditor zu erstellen, zu kompilieren und zur Ausführung zu bringen</li></ul>
<b>Inhalt:</b>
<ul style="list-style-type: none"><li>• Bildung von Teams/Kleingruppen, Soziale Vernetzung</li><li>• Kennenlernen wichtiger Anlaufstellen und Personen am Campus</li><li>• Einarbeitung in die Programmierumgebung Scratch</li><li>• Erstellung eines Profils für alle Teammitglieder in Form einer Scratch-Applikation</li><li>• Installation des Virtualisierungssystems VirtualBox und einer virtuellen Maschine, die im Rahmen von Vorlesungen des ersten und zweiten Semesters verwendet wird</li><li>• Einführung in die grundlegende Bedienung der virtuellen Maschine</li><li>• Verfassen einfacher Konfigurationsdateien</li><li>• Erstellen und Ausführung einfacher Programmdateien</li><li>• Bibliothekseinführung mit Online-Test</li></ul>
<b>Literatur:</b>
<ul style="list-style-type: none"><li>• MÖLLER, Bernhard. <i>Erstelle Geschichten, Spiele und Animationen</i> [online]. [Zugriff am: 30.08.2023]. Verfügbar unter: <a href="https://medienkompass.de/was-ist-scratch-und-wie-kann-man-damit-spiele-programmieren/">https://medienkompass.de/was-ist-scratch-und-wie-kann-man-damit-spiele-programmieren/</a></li></ul>
<b>Anmerkungen:</b>
Keine Anmerkungen

<b>Grundlagen der Programmierung 1</b>			
<b>Modulkürzel:</b>	IB_GP1	<b>SPO-Nr.:</b>	2
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	1
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Wintersemester
<b>Modulverantwortliche(r):</b>	Regensburger, Franz		
<b>Leistungspunkte / SWS:</b>	7 ECTS / 6 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	70 h	
	Selbststudium:	105 h	
	Gesamtaufwand:	175 h	
<b>Lehrveranstaltungen des Moduls:</b>	2.1: Grundlagen der Programmierung 1 2.2: Praktikum Grundlagen der Programmierung 1		
<b>Lehrformen des Moduls:</b>	2.1: SU/Ü - seminaristischer Unterricht/Übung 2.2: Pr - Praktikum		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
2.1: schrP90 - schriftliche Prüfung, 90 Minuten 2.2: LN - praktische Prüfung/Arbeit			
Weitere Erläuterungen: Im Rahmen des Praktikums müssen mehrere Testate (Programmieraufgaben in C) erworben werden. Bei erfolgreicher Bearbeitung der Aufgabenstellung wird vom Dozenten jeweils ein Testat vergeben. Die Lösungen dürfen und sollen zur Förderung der sozialen und fachlichen Kompetenz in Kleingruppen erarbeitet werden. Insgesamt müssen vier Aufgaben bearbeitet werden, die wesentliche Themen der Vorlesung behandeln. Die fertigen Lösungen sind einzeln innerhalb eines festen Terminrasters (alle 14 Tage ein Testat) individuell von den Teilnehmern zu präsentieren, wobei auch Fragen zum Lösungskonzept und zum erstellten Programm zu beantworten sind. Nur wenn alle vier Testate rechtzeitig erworben werden, gilt der Leistungsnachweis als erbracht.			
<b>Voraussetzungen gemäß SPO:</b>			
Keine			
<b>Empfohlene Voraussetzungen:</b>			
Sicherer Umgang mit dem eigenen PC (Anforderungen Intel/AMD 64, 16GB RAM, 500GB SSD, OS Linux/MacOS/Windows).			
Den Studierenden wird ein gebrauchsfertiges Image einer virtuellen Maschine für das Selbststudium zuhause zur Verfügung gestellt, welches unter allen Intel/AMD Plattformen mittels VirtualBox oder anderer gängiger Hypervisor zur Ausführung gebracht werden kann.			
Des Weiteren wird dieses Image in den PC-Pools der Fakultät zur Verfügung gestellt.			

**Angestrebte Lernergebnisse:**

Nach dem Besuch der Vorlesung sind die Studierenden in der Lage,

- allgemeine Begriffe der Informatik zu nennen und zu definieren
- in Grundzügen die historische Entwicklung von Programmiersprachen zu beschreiben
- einfache Probleme logisch zu erfassen und selbständig eine algorithmische Lösung zu erstellen
- in einer höheren imperativen Programmiersprache vorgegebene oder selbst entwickelte Algorithmen zu implementieren, insbesondere in C
- Dienste des Betriebssystems und einer Entwicklungsumgebung zu nutzen
- allein oder gemeinsam in kleinen Teams die Lösung von Programmieraufgaben zu diskutieren und zu erstellen

Nach dem Besuch des Praktikums sind die Studierenden in der Lage,

- vorgegebene Code-Teile zu verstehen und selbständig Erweiterungen im Code vorzunehmen
- umfangreichere C-Programme (zwischen 500 - 2000 Zeilen Code) zu erstellen
- die wesentlichen Komponenten einer Entwicklungsumgebung (Editor, Compiler, Debugger und Build-Tool) zu bedienen
- allein oder gemeinsam in kleinen Teams Programmieraufgaben zu lösen
- eine Linux Workstation (virtuelle Maschine) zu bedienen
- das Werkzeug GIT für die Versionsverwaltung des Quellcodes mit Repositories auf dem PC und in der Cloud zu nutzen

**Inhalt:**

Vorlesung:

- Grundbegriffe der Informatik, Phasen und Werkzeuge der Software-Entwicklung, Grundbegriffe und Prinzipien der imperativen Programmierung
- Programmiersprachen (allgemein und speziell Sprache C)
- Ablaufsteuerung, primitive Datentypen in C
- Getrennte Übersetzung und Entwicklungsumgebung (Editor, Build-Tool, Debugger)
- Enumerationen und Datentyp bool
- Funktionen, Unterprogrammtechnik, Parameterübergabe, Auf- und Abbau des Stacks
- Records
- Arrays
- Pointer
- Statische und dynamische Speicherobjekte, Gültigkeit, Sichtbarkeit und Lebensdauer
- Verkettete Listen und andere Speichergeflechte
- String-Funktionen der Standardbibliothek

Praktikum:

- Erstellung eines interaktiven Spiels (Worm) mit einfacher Symbolgrafik auf Basis der Curses-Bibliothek
- Arbeiten mit einer einfachen, nicht-integrierten Entwicklungsumgebung in der Konsole (Editor, Compiler, Debugger)
- Nutzung einer Linux-Workstation für die Entwicklung von C-Programmen
- Arbeiten in einer Remote-Sitzung per SSH
- Verwendung eines einfachen Build-Tools (GNU make)
- Verwendung des RCS GIT mit Repositories auf dem eigenen Rechner und in der Cloud
- Die Programmierung in der Sprache C erfolgt auf Basis einer virtuellen Linux-Maschine, deren Image in allen Rechner-Pools der Fakultät vorinstalliert ist. Dieses Image kann weiterhin von allen Studierenden kopiert werden und auf dem eigenen PC genutzt werden. In der virtuellen Maschine wird ausschließlich OpenSource-Software verwendet, so dass das Image der virtuellen Maschine beliebig oft kopiert und weitergegeben werden darf.

**Literatur:**

- DAUSMANN, Manfred und andere, 2011. *C als erste Programmiersprache: vom Einsteiger zum Fortgeschrittenen ; [mit Übungsaufgaben im Web]*. 7. Auflage. Wiesbaden: Vieweg + Teubner. ISBN 978-3-8348-1221-6, 3-8348-1221-8
- ERNST, H., 2008. *Grundkurs Informatik. Grundlagen und Konzepte für die erfolgreiche IT-Praxis*. 4. Auflage. ISBN 978-3-8348-0362-7
- ERBEN, Thomas, 2017. *Einführung in Unix/Linux für Naturwissenschaftler: Effizientes wissenschaftliches Arbeiten mit der Unix-Kommandozeile* [online]. Berlin, Heidelberg: Springer Berlin Heidelberg PDF eBook. ISBN 978-3-662-50301-0. Verfügbar unter: <https://doi.org/10.1007/978-3-662-50301-0>.
- FUCHS, Paul, 2021. *Git Handbuch für Einsteiger: der leichte Weg zum Git-Experten*. 1. Auflage. [Deggen-dorf] ; [Berlin]: BMU Verlag. ISBN 978-3-96645-119-2, 3-96645-119-0

**Anmerkungen:**

Keine Anmerkungen

<b>Rechnerarchitektur</b>			
<b>Modulkürzel:</b>	IB_RA	<b>SPO-Nr.:</b>	3
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	1
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Wintersemester
<b>Modulverantwortliche(r):</b>	Hagerer, Andreas		
<b>Leistungspunkte / SWS:</b>	7 ECTS / 5 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	58 h	
	Selbststudium:	117 h	
	Gesamtaufwand:	175 h	
<b>Lehrveranstaltungen des Moduls:</b>	3.1: Rechnerarchitektur 3.2: Praktikum Rechnerarchitektur		
<b>Lehrformen des Moduls:</b>	3.1: SU/Ü - seminaristischer Unterricht/Übung 3.2: Pr - Praktikum		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
3.1: schrP90 - schriftliche Prüfung, 90 Minuten 3.2: LN - ohne/mit Erfolg teilgenommen  Weitere Erläuterungen: Voraussetzung für die Teilnahme an der schriftlichen Prüfung ist ein erfolgreich abgeschlossenes Praktikum. Im Rahmen des Praktikums müssen mehrere Testate erworben werden. Bei erfolgreicher Bearbeitung der Aufgabenstellung wird vom Dozenten jeweils ein Testat vergeben. Insgesamt müssen mindestens 80% der Testate bearbeitet werden, die wesentliche Themen der Vorlesung behandeln.			
<b>Voraussetzungen gemäß SPO:</b>			
Keine			
<b>Empfohlene Voraussetzungen:</b>			
Grundkenntnisse der Booleschen Algebra und algorithmisches Denken			
<b>Angestrebte Lernergebnisse:</b>			
Nach erfolgreicher Teilnahme an den Modulveranstaltungen sind die Studierenden in der Lage, <ul style="list-style-type: none"> <li>• Informationen unterschiedlicher Art für eine Verarbeitung durch digitale Rechner darzustellen</li> <li>• ausgehend von der datenverarbeitungsgerechten Darstellung von Information und den Prinzipien der befehlsbasierten Ausführung von Verarbeitungsvorschriften die Elemente einer Befehlssatzarchitektur zu benennen und ihre Auswirkungen auf die Programmierung zu beurteilen</li> <li>• das Zusammenspiel von Hardware und Software zu erklären</li> <li>• grundlegende Elemente prozeduraler Programmierung in einer Maschinsprache zu formulieren</li> <li>• die Wirkung von Programmierungsalternativen auf die Ausführungsgeschwindigkeit zu beurteilen</li> <li>• Konzepte zur Leistungssteigerung in modernen Prozessoren und die damit verbundenen Problematiken zu erläutern</li> </ul>			



Praktikum: <ul style="list-style-type: none"> <li>• selbständig kleine Assemblerprogramme zu erstellen und zur Ausführung zu bringen</li> </ul>
<b>Inhalt:</b>
<ul style="list-style-type: none"> <li>• Klassifikation von Rechnerarchitekturen und Rechnerbeschreibung</li> <li>• Informationsdarstellung im Rechner: Polyadische Zahlensysteme, Konvertierungsverfahren, Komplementdarstellungen, Gleitkommadarstellungen</li> <li>• Befehlssatzarchitekturen: Klassifikation, Programmiermodell, IA32-Befehlssatz als Beispiel, maschinen-nahe Programmierkonzepte (Adressierungsarten, Unterprogrammtechnik, andere)</li> <li>• von Neumann-Rechner: Arbeitsweise und Struktur, Leitwerk, Rechenwerk, Speicherwerk und Speichertechnologien, E/A-Werke, Bussystem</li> <li>• Fortgeschrittene Konzepte in heutigen Rechnerarchitekturen (Cache, Mehrkern-Architekturen, Pipelining, Superskalarität)</li> </ul> Praktikum: <ul style="list-style-type: none"> <li>• Aufgaben zur Assemblerprogrammierung mit NASM (x86/IA32/Intel 64-Assembler)</li> </ul>
<b>Literatur:</b>
<ul style="list-style-type: none"> <li>• PATTERSON, David A. und John L. HENNESSY, 2014. <i>Computer Organization and Design: The Hardware/Software Interface</i>. 5. Auflage.</li> <li>• SCHIFFMANN, W. und R. SCHMITZ, 2003. <i>Technische Informatik 1 – Grundlagen der digitalen Elektronik</i>. 5. Auflage.</li> <li>• SCHIFFMANN, W. und R. SCHMITZ, 2002. <i>Technische Informatik 2 – Grundlagen der Computertechnik</i>. 4. Auflage.</li> <li>• STALLINGS, William, 2017. <i>Computer Organization and Architecture - Designing for Performance</i>. 10. Auflage.</li> <li>• ROMING, Marcus und Joachim ROHDE, 2006. <i>Assembler : Grundlagen der Programmierung</i>. ISBN 978-3-8266-1469-9</li> </ul>
<b>Anmerkungen:</b>
Keine Anmerkungen

<b>Mathematische Grundlagen 1</b>			
<b>Modulkürzel:</b>	IB_MGI1	<b>SPO-Nr.:</b>	4
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	1
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Wintersemester
<b>Modulverantwortliche(r):</b>	Lorencka, Joanna		
<b>Leistungspunkte / SWS:</b>	6 ECTS / 5 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	59 h	
	Selbststudium:	91 h	
	Gesamtaufwand:	150 h	
<b>Lehrveranstaltungen des Moduls:</b>	4.1: Mathematische Grundlagen 1 4.2: Übung zu Mathematische Grundlagen 1		
<b>Lehrformen des Moduls:</b>	4.1: SU - seminaristischer Unterricht 4.2: Ü - Übung		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
4.1: schrP90 - schriftliche Prüfung, 90 Minuten 4.2: LN - ohne Leistungsnachweis			
Weitere Erläuterungen: Keine			
<b>Voraussetzungen gemäß SPO:</b>			
Keine			
<b>Empfohlene Voraussetzungen:</b>			
Sichere Umgang mit Potenzgesetzen und algebraische Umformungen, Kenntnisse über Funktionen und ihre Eigenschaften			
<b>Angestrebte Lernergebnisse:</b>			
Nach erfolgreicher Teilnahme an der Lehrveranstaltung sind die Studierenden in der Lage, <ul style="list-style-type: none"> <li>• Beweisstrukturen zu verstehen und informatikrelevante Beweise durchzuführen</li> <li>• Grundlagen der Algebra und Logik wiederzugeben und auf fachspezifische Aufgaben anzuwenden</li> <li>• mit Matrizen zu rechnen, beispielsweise um lineare Gleichungssystemen zu lösen</li> <li>• komplexe Zahlen in unterschiedliche Formen darzustellen, um Gleichungen und Ungleichungen zu lösen</li> <li>• Grenzwertprozesse zu analysieren</li> <li>• Formel und Sätze aus der Differentialrechnung wiederzugeben, anzuwenden und zu interpretieren</li> <li>• Approximationsverfahren verstehen und auf fachspezifische Aufgaben anzuwenden</li> <li>• Taylorpolynome zu entwickeln und den Fehler, der durch die Polynomdarstellung entsteht, mit Hilfe des Lagrangeschen Restglieds abzuschätzen</li> <li>• Funktionen als Lösungen zu Differentialgleichungen nachzuprüfen.</li> </ul>			

<ul style="list-style-type: none"><li>• homogene Differentialgleichungen mit der Exponentialansatz zu lösen</li></ul>
<b>Inhalt:</b>
<ul style="list-style-type: none"><li>• Abbildungen, Logische Schaltungen, Aussagenlogik, elementare Mengenlehre, Binärwörter, Binomialkoeffizienten, Boolesche Algebra, Quantorenlogik</li><li>• Folgen und Reihen</li><li>• Komplexe Zahlen</li><li>• Matrizenkalkül</li><li>• Lineare Gleichungssysteme</li><li>• Differentialrechnung</li><li>• Einführung in Differentialgleichungen</li></ul>
<b>Literatur:</b>
<ul style="list-style-type: none"><li>• ERVEN, J., 2011. <i>Taschenbuch der Ingenieurmathematik</i>.</li><li>• TESCHL, G. und S. TESCHL, 2008. <i>Mathematik für Informatiker, Bd. 1</i>.</li><li>• HARTMANN, Peter, 2015. <i>Mathematik für Informatiker: ein praxisbezogenes Lehrbuch</i> [online]. Wiesbaden: Springer Vieweg PDF e-Book. ISBN 978-3-658-03415-3, 978-3-658-03416-0. Verfügbar unter: <a href="https://doi.org/10.1007/978-3-658-03416-0">https://doi.org/10.1007/978-3-658-03416-0</a>.</li><li>• TESCHL, Gerhard und Susanne TESCHL, 2008. <i>Mathematik für Informatiker, Bd. 2</i>.</li><li>• KEMNITZ, Arnfried, 2019. <i>Mathematik zum Studienbeginn: Grundlagenwissen für alle technischen, mathematisch-naturwissenschaftlichen und wirtschaftswissenschaftlichen Studiengänge</i> [online]. Wiesbaden: Springer Spektrum PDF e-Book. ISBN 978-3-658-26604-2. Verfügbar unter: <a href="https://doi.org/10.1007/978-3-658-26604-2">https://doi.org/10.1007/978-3-658-26604-2</a>.</li></ul>
<b>Anmerkungen:</b>
Keine Anmerkungen

<b>Physikalische und Elektrotechnische Grundlagen</b>			
<b>Modulkürzel:</b>	IB_PEG	<b>SPO-Nr.:</b>	5
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	1
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Wintersemester
<b>Modulverantwortliche(r):</b>	Huber, Siegfried		
<b>Leistungspunkte / SWS:</b>	5 ECTS / 4 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	47 h	
	Selbststudium:	78 h	
	Gesamtaufwand:	125 h	
<b>Lehrveranstaltungen des Moduls:</b>	Physikalische und Elektrotechnische Grundlagen		
<b>Lehrformen des Moduls:</b>	SU/Ü - seminaristischer Unterricht/Übung		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
schrP90 - schriftliche Prüfung, 90 Minuten			
Weitere Erläuterungen: Keine			
<b>Voraussetzungen gemäß SPO:</b>			
Keine			
<b>Empfohlene Voraussetzungen:</b>			
Interesse an physikalischen Grundlagen der Informatik			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch des Moduls sind die Studierenden in der Lage,			
<ul style="list-style-type: none"> <li>• physikalisches und elektrotechnisches Grundwissen im Kontext der Informatik anzuwenden</li> <li>• die wichtigsten Phänomene und Gesetzmäßigkeiten aus der Elektrotechnik zu verstehen und diese auf technische Vorgänge mit der nötigen Sicherheit anzuwenden,</li> <li>• grundlegende physikalische Zusammenhänge, insbesondere aus der Sicht der Informatik, zu erkennen und sie zur praktischen Umsetzung auf Probleme der Informatik zu nutzen</li> <li>• die grundlegende Funktion und Aufgabe von Widerständen, Spulen und Kondensatoren zu erklären</li> <li>• die grundlegende Funktion und Aufgabe von Dioden und Transistoren zu erklären</li> <li>• einfache digitale Schaltungen wie FlipFlops zu erklären</li> </ul>			
<b>Inhalt:</b>			
<ul style="list-style-type: none"> <li>• Physikalische Grundlagen (Kräfte, Ladung, Aufbau der Materie)</li> <li>• Gleichstrom</li> <li>• Elektrisches Feld</li> <li>• magnetisches Feld</li> </ul>			

- Wechselstromanwendungen
- Grundlagen der Schaltungs- und Digitaltechnik

**Literatur:**

- STUART, Herbert A. und Gerhard KLAGES, 2009. *Kurzes Lehrbuch der Physik: mit 22 Tabellen, 235 Aufgaben und ausführlichen Lösungen*. 19. Auflage. Berlin [u.a.]: Springer. ISBN 978-3-540-89045-4, 978-3-540-89046-1
- MESCHEDE, Dieter, GERTHSEN, Christian, 2015. *Gerthsen Physik* [online]. Berlin [u.a.]: Springer Spektrum PDF e-Book. ISBN 978-3-662-45977-5, 978-3-662-45976-8. Verfügbar unter: <https://doi.org/10.1007/978-3-662-45977-5>.
- LINSE, Hermann, FISCHER, Rolf, 2005. *Elektrotechnik für Maschinenbauer: Grundlagen und Anwendungen ; mit 25 Tabellen und 120 Beispielen* [online]. Wiesbaden: Teubner PDF e-Book. ISBN 978-332-29278-1-1. Verfügbar unter: <https://doi.org/10.1007/978-3-322-92781-1>.

**Anmerkungen:**

Keine Anmerkungen

<b>Englisch</b>			
<b>Modulkürzel:</b>	IB_ENG	<b>SPO-Nr.:</b>	6
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	1
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Wintersemester
<b>Modulverantwortliche(r):</b>	Reicherstorfer, Anja		
<b>Leistungspunkte / SWS:</b>	3 ECTS / 2 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	24 h	
	Selbststudium:	51 h	
	Gesamtaufwand:	75 h	
<b>Lehrveranstaltungen des Moduls:</b>	Englisch		
<b>Lehrformen des Moduls:</b>	SU/Ü - seminaristischer Unterricht/Übung		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
schrP90 - schriftliche Prüfung, 90 Minuten			
Weitere Erläuterungen: Keine			
<b>Voraussetzungen gemäß SPO:</b>			
Keine			
<b>Empfohlene Voraussetzungen:</b>			
Grundlegende Kommunikationsfähigkeit in englischer Sprache, entsprechend der Schulausbildung, die für die Erlangung der Hochschulzugangsberechtigung eingebracht wurde			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch des Moduls sind die Studierenden in der Lage, <ul style="list-style-type: none"> <li>• ihren Wortschatz zu erweitern, insbesondere im Bereich IT</li> <li>• ihre Schreib- und Sprechfertigkeiten durch geeignete Simulation und die situationsbezogene Anwendung der englischen Sprache zu verbessern</li> <li>• ihre Kenntnisse der englischen Grammatik zu vertiefen</li> <li>• durch Diskussion ausgewählter Fachthemen ihre Kommunikationsfähigkeit und ihr Sprachgefühl zu verbessern</li> <li>• selbständig den Inhalt englischer Fachtexte zu erschließen</li> <li>• englische Fachzeitschriften und Bücher im Rahmen von Recherchen und Programmierprojekten zu nutzen</li> </ul>			
<b>Inhalt:</b>			
<ul style="list-style-type: none"> <li>• Ausgewählte IT-Themen wie z.B. computer users, computer architecture, computer applications, operating systems, multimedia, networks, the World Wide Web, webpage creator, communication systems,</li> </ul>			

computing support, data security, software engineering, people in computing, the future of IT, electronic publishing

- Referatsthemen aus der aktuellen englischsprachigen Presse
- Übungen zu Grammatik, Semantik, Hörverstehen, Präsentation von Texten, Einführung in die geschäftliche Korrespondenz
- Verwendung von Fachbüchern zum Selbststudium und zur Weiterbildung
- Umgang mit Fachartikeln und wissenschaftlichen Publikationen
- Verwendung von Online-Ressourcen, wie z.B. stackoverflow (<https://stackoverflow.com/>) im Rahmen von Programmierprojekten
- Nutzung von Online-Foren für Fragen und Beiträge in englischer Sprache

**Literatur:**

- GLENDINNING, Eric H. und John MACEWAN, . *Oxford English for information technology*. Oxford [u.a.]: Oxford Univ. Press.
- NATIONAL UNIVERSITY LIBRARY, . *Reading a scientific article* [online]. [Zugriff am: 30.8.2023]. Verfügbar unter: <https://resources.nu.edu/researchprocess/readingscientificarticle>
- NATIONAL UNIVERSITY LIBRARY, . *Evaluating Information* [online]. [Zugriff am: 30.8.2023]. Verfügbar unter: <https://resources.nu.edu/researchprocess/evaluatinginfo>

**Anmerkungen:**

Keine Anmerkungen

<b>Grundlagen der Programmierung 2</b>			
<b>Modulkürzel:</b>	IB_GP2	<b>SPO-Nr.:</b>	7
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	2
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Sommersemester
<b>Modulverantwortliche(r):</b>	Apel, Sebastian		
<b>Leistungspunkte / SWS:</b>	7 ECTS / 6 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	70 h	
	Selbststudium:	105 h	
	Gesamtaufwand:	175 h	
<b>Lehrveranstaltungen des Moduls:</b>	7.1: Grundlagen der Programmierung 2 7.2: Praktikum Grundlagen der Programmierung 2		
<b>Lehrformen des Moduls:</b>	7.1: SU/Ü - seminaristischer Unterricht/Übung 7.2: Pr - Praktikum		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
7.1: schrP90 - schriftliche Prüfung, 90 Minuten 7.2: LN - ohne/mit Erfolg teilgenommen			
<p>Weitere Erläuterungen:</p> <p>Im Rahmen des Praktikums müssen mehrere Testate (Programmieraufgaben in Java) erworben werden. Bei erfolgreicher Bearbeitung der Aufgabenstellung wird vom Dozenten jeweils ein Testat vergeben.</p> <p>Insgesamt müssen fünf Aufgaben bearbeitet werden, die wesentliche Themen der Vorlesung behandeln. Die fertigen Lösungen sind einzeln zu präsentieren, wobei auch Fragen zum Lösungskonzept und zum erstellten Programm zu beantworten sind.</p> <p>Nur wenn alle fünf Aufgaben rechtzeitig vorgeführt werden, gilt der Leistungsnachweis als erbracht.</p>			
<b>Voraussetzungen gemäß SPO:</b>			
Keine			
<b>Empfohlene Voraussetzungen:</b>			
Das Modul 'Grundlagen der Programmierung 1' wurde bereits besucht und dort vermittelte Kompetenzen in der prozeduralen Programmierung liegen vor			
<b>Angestrebte Lernergebnisse:</b>			
<p>Nach dem Besuch des Moduls sind die Studierenden in der Lage,</p> <ul style="list-style-type: none"> <li>• die wesentlichen Merkmale einer objektorientierten Programmiersprache anzugeben und zu erklären</li> <li>• die grundlegenden Programmierkonstrukte von Java in konkreten Programmen einzusetzen</li> <li>• die wichtigsten Funktionen einer modernen Entwicklungsumgebung zu erklären und zur Programmerstellung zu nutzen</li> <li>• die Problematik des Nachweises der Korrektheit eines Programms zu erklären und eine systematische Testmethodik zur Fehlerreduktion zu beschreiben und anzuwenden</li> </ul>			



- das Werkzeug GIT zur Verwaltung des eigenen Quellcodes auf dem eigenen PC zu nutzen und dessen Datenbank mit einem Repository in der Cloud zu synchronisieren
- die praktische Anwendung der gelernten Inhalte in einer mittelgroßen Programmieraufgabe erfolgreich nachzuweisen (=Praktikum)

Praktikum: Studierende sind in der Lage, objektorientierte Programmier Techniken anzuwenden und eine vollständige Anwendung mit interaktiver grafischer Oberfläche zu erstellen.

**Inhalt:**

- Einführung: Objektorientierung und Java
- Entwicklungsumgebung und Laufzeitumgebung
- Speicherverwaltung, Referenzen
- Klassen, Vererbung, Polymorphie
- Container-Klassen: Collections und Generics
- Fortgeschrittene Sprachkonzepte: Interfaces, Exceptions, Packages
- Graphische Benutzeroberflächen, Umgang mit asynchronen Ereignissen
- Nebenläufigkeit

Im Rahmen des Praktikums wird ein Media-Player mit grafischer Oberfläche implementiert.

**Literatur:**

- KRÜGER, Guido und Heiko HANSEN, 2014. *Java-Programmierung - das Handbuch zu Java 8*. 8. Auflage.
- ULLENBOOM, Christian, 2016. *Java ist auch eine Insel -- Einführung, Ausbildung, Praxis*. 12. Auflage.
- ABTS, Dietmar, 2016. *Grundkurs JAVA -- Von den Grundlagen bis zu Datenbank- und Netzanwendungen*. 9. Auflage.

**Anmerkungen:**

Keine Anmerkungen

<b>Mathematische Grundlagen 2</b>			
<b>Modulkürzel:</b>	IB_MGI2	<b>SPO-Nr.:</b>	8
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	2
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Sommersemester
<b>Modulverantwortliche(r):</b>	Lorencka, Joanna		
<b>Leistungspunkte / SWS:</b>	6 ECTS / 5 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	59 h	
	Selbststudium:	91 h	
	Gesamtaufwand:	150 h	
<b>Lehrveranstaltungen des Moduls:</b>	8.1: Mathematische Grundlagen 2 8.2: Übung zu Mathematische Grundlagen 2		
<b>Lehrformen des Moduls:</b>	8.1: SU - seminaristischer Unterricht 8.2: Ü - Übung		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
8.1: schrP90 - schriftliche Prüfung, 90 Minuten 8.2: LN - ohne Leistungsnachweis  Weitere Erläuterungen: Keine			
<b>Voraussetzungen gemäß SPO:</b>			
keine			
<b>Empfohlene Voraussetzungen:</b>			
Mathematische Grundlagen 1			
<b>Angestrebte Lernergebnisse:</b>			
<p>Nach erfolgreicher Teilnahme an den Modulveranstaltungen sind die Studierenden in der Lage,</p> <ul style="list-style-type: none"> <li>• zwischen Gruppen, Ringen- und Körperstrukturen zu unterscheiden und anhand der modularen Arithmetik zu verdeutlichen</li> <li>• die Grundlagen der linearen Algebra und die Eigenschaften linearer Abbildungen zu beschreiben</li> <li>• Eigenwerte und Eigenvektoren zu berechnen und Matrizen zu diagonalisieren</li> <li>• analytische Funktionen in Potenzreihen zu entwickeln.</li> <li>• die Definition des Riemann Integrals, den HDI und den Mittelwertsatz der Integralrechnung wiederzugeben</li> <li>• die Integrationstechniken Substitution, partielle Integration, Integration über Partialbruchzerlegung und Potenzreihenentwicklung zu verstehen und anzuwenden</li> <li>• grundlegende Konzepte der Numerik in Zusammenhang mit Integration wiederzugeben und diese anzuwenden</li> </ul>			

<ul style="list-style-type: none"><li>• Differentialgleichungen erster Ordnung durch Trennung der Variablen, integrierende Faktoren und Variation der Parameter zu lösen</li></ul>
<b>Inhalt:</b>
<ul style="list-style-type: none"><li>• Determinanten</li><li>• Eigenwerte und Eigenvektoren</li><li>• Injektive, surjektive und bijektive Abbildungen</li><li>• modulare Arithmetik mit Anwendungen</li><li>• Integralrechnung</li><li>• Potenzreihen</li><li>• Einführung in die numerische Mathematik</li></ul>
<b>Literatur:</b>
<ul style="list-style-type: none"><li>• TESCHL, G. und S. TESCHL, 2008. <i>Mathematik für Informatiker, Bd. 1</i>. ISBN <a href="https://doi.org/10.1007/978-3-642-37972-7">https://doi.org/10.1007/978-3-642-37972-7</a></li><li>• ERVEN, J., 2011. <i>Taschenbuch der Ingenieurmathematik</i>.</li><li>• KREYSZIG, E., 1993. <i>Advanced Engineering Mathematics</i>.</li><li>• TESCHL, Gerhard und Susanne TESCHL, 2008. <i>Mathematik für Informatiker, Bd. 2</i>. ISBN <a href="https://doi.org/10.1007/978-3-642-54274-9">https://doi.org/10.1007/978-3-642-54274-9</a></li></ul>
<b>Anmerkungen:</b>
Keine Anmerkungen

<b>Betriebssysteme</b>			
<b>Modulkürzel:</b>	IB_BS	<b>SPO-Nr.:</b>	9
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	2
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Sommersemester
<b>Modulverantwortliche(r):</b>	Regensburger, Franz		
<b>Leistungspunkte / SWS:</b>	5 ECTS / 4 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	47 h	
	Selbststudium:	78 h	
	Gesamtaufwand:	125 h	
<b>Lehrveranstaltungen des Moduls:</b>	Betriebssysteme		
<b>Lehrformen des Moduls:</b>	SU/Ü - seminaristischer Unterricht/Übung		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
schrP90 - schriftliche Prüfung, 90 Minuten			
Weitere Erläuterungen: Keine			
<b>Voraussetzungen gemäß SPO:</b>			
Keine			
<b>Empfohlene Voraussetzungen:</b>			
Das Modul 'Grundlagen der Programmierung 1' wurde besucht. Solide Kenntnisse der Sprache C sind vorhanden.			
<b>Angestrebte Lernergebnisse:</b>			
<p>Nach dem Besuch des Moduls sind die Studierenden in der Lage,</p> <ul style="list-style-type: none"> <li>• die grundlegenden Aufgaben eines Betriebssystems zu erklären</li> <li>• die grundlegenden Architekturen gängiger Betriebssysteme wie Linux, Unix und Windows zu unterscheiden</li> <li>• die Systemprogrammierschnittstelle POSIX-konformer Systeme (SUS) zu nutzen</li> <li>• unterschiedliche Ausprägungen der Parallelverarbeitung zu benennen</li> <li>• die wesentlichen Komponenten der Prozessmodelle der Betriebssysteme Linux und Windows zu benennen</li> <li>• den Unterschied zwischen Prozessen und Threads zu beschreiben und können deren Einsatzmöglichkeiten sowie deren Vor- und Nachteile erklären</li> <li>• mehrere Schedulingstrategien zu erklären und können diese in einem abstrakten Modell anwenden</li> <li>• die Funktionsweise der virtuellen Speicherverwaltung zu erklären und können unterschiedliche Strategien und Algorithmen zur Speicherverwaltung in abstrakten Modellen anwenden</li> </ul>			

- die Probleme, die beim konkurrierenden Zugriff auf gemeinsame Ressourcen auftreten, zu erklären und wissen, wie einschlägige Synchronisationsmechanismen zur Verhinderung dieser Probleme eingesetzt werden können
- einfache Mechanismen zu benutzen, mit deren Hilfe Prozesse kommunizieren können
- die Dienste eines Dateisystems zu benennen und können diese anhand eines einfachen Dateisystems erklären

**Inhalt:**

- Einführung: Aufgaben eines Betriebssystems
- Grundlagen: Systemprogrammierung
- Prozesse und Threads: Parallelverarbeitung, Prozessmodell, Threads, Scheduling (Algorithmen mit Zeitscheiben und Prioritäten)
- Speicherverwaltung (\*-Fit, Buddy-Algorithmus, LRU, Clock-Algorithmus)
- Synchronisation von Prozessen und Threads (Semaphore, Mutexe)
- Kommunikation von Prozessen und Threads (Pipes, Sockets)
- Dateisystem (FAT)

**Literatur:**

- GLATZ, Eduard, 2019. *Betriebssysteme: Grundlagen, Konzepte, Systemprogrammierung*. 4. Auflage. ISBN 978-3-86490-705-0
- KERRISK, Michael, 2010. *The Linux Programming Interface*. ISBN 1-59327-220-0
- TANNENBAUM, Andrew S., 2016. *Moderne Betriebssysteme*. 4. Auflage. ISBN 978-3-86894-270-5
- BOVET, und CESATI, 2005. *Understanding the Linux Kernel*. 3. Auflage. ISBN 0-596-00565-2
- ROCHKIND, Marc J., 2004. *Advanced UNIX Programming*. 2. Auflage. ISBN 0-13-141154-3

**Anmerkungen:**

Die Beispiele der Vorlesung, Demonstrationen am Rechner sowie die Übungsaufgaben werden auf Basis einer virtuellen Linux-Maschine ausgeführt, deren Image in allen Rechner-Pools der Fakultät vorinstalliert ist. Dieses Image kann weiterhin von allen Studierenden kopiert werden und auf dem eigenen PC genutzt werden.

In der virtuellen Maschine wird ausschließlich OpenSource-Software verwendet, so dass das Image der virtuellen Maschine beliebig oft kopiert und weitergegeben werden darf.

Das Image enthält auch Software für die höheren Semester, so dass die virtuelle Linux-Maschine während des gesamten Studiums genutzt werden kann.

<b>Technik der Mensch-Maschine Interaktion</b>			
<b>Modulkürzel:</b>	IB_TMMI	<b>SPO-Nr.:</b>	10
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	2
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Sommersemester
<b>Modulverantwortliche(r):</b>	Glavina, Bernhard		
<b>Leistungspunkte / SWS:</b>	5 ECTS / 4 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	47 h	
	Selbststudium:	78 h	
	Gesamtaufwand:	125 h	
<b>Lehrveranstaltungen des Moduls:</b>	Technik der Mensch-Maschine Interaktion		
<b>Lehrformen des Moduls:</b>	SU/Ü - seminaristischer Unterricht/Übung		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
schrP90 - schriftliche Prüfung, 90 Minuten			
Weitere Erläuterungen:			
Für Teilnehmer aus UXD gilt: Das zugehörige Praktikum UXD_TMIP muss erfolgreich abgelegt sein, damit ein Antritt bei der Vorlesungsprüfung möglich ist. Für weitere Informationen kontaktieren Sie bitte den Studiengangleiter UXD (Prof. Dr. Andreas Riener).			
<b>Voraussetzungen gemäß SPO:</b>			
Keine			
<b>Empfohlene Voraussetzungen:</b>			
Vertrautheit mit der Benutzung gängiger UI-Technologien (Smartphone, Tablet)			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch des Moduls sind die Studierenden in der Lage,			
<ul style="list-style-type: none"> <li>• die Terminologie von Mensch-Maschine Interaktion und deren Grundbegriffe zu erläutern sowie Probleme zu erklären</li> <li>• eine Abgrenzung zu ähnlichen Themen zu machen</li> <li>• die historische Entwicklung von Mensch-Maschine Interaktion (MMI) zu beschreiben und können Erkenntnisse aus der Vergangenheit auf zukünftige Entwicklungen anwenden</li> <li>• unterschiedlichste 2D Ein- und Ausgabegeräte für die Mensch-Maschine Interaktion aufzuzählen und miteinander zu vergleichen</li> <li>• sowohl Potential als auch Einsatzprobleme von 3D Ein- und Ausgabegeräten wiederzugeben und zu bewerten</li> <li>• für konkrete Aufgabenstellungen beurteilen zu können, welche Hardware/Technologie für die Lösung am geeignetsten ist</li> </ul>			

**Inhalt:**

In dieser Lehrveranstaltung werden die Geschichte, Gegenwart und Zukunft von Mensch-Maschine-Interaktion anhand von Literatur und Beispielen beleuchtet.

- Einführung und Motivation: Grundbegriffe/-verständnis und Terminologie von Mensch-Maschine Interaktion, Probleme des Fachgebiets, Geschichtlicher Überblick
- 2D Eingabegeräte: Gestaltungsanforderungen für Eingabegeräte (ISO 9241-410), Wertgeber, Tastaturen, Maus, Lichtgriffel, Touchscreen-Technologien (optisch, akustisch, kapazitiv, resistiv, induktiv), Fat-Finger und Midas-Touch Problem, Handschriftliche Eingabe, Optische Eingabe (Kameras, Scanner, Eye Tracking), Akustische Eingabe (Mikrofone, Spracherkennung/-steuerung), Haptische/Taktile Eingabe, Bio- und Neurosensoren
- 2D Ausgabegeräte: Visuelle Displays (visuelle Wahrnehmung, curved/tiled Displays, Edge Blending), Auditive Displays (auditive Wahrnehmung, räumliches Hören, Earcons/Lyricons), Haptische Displays (taktile Wahrnehmung, Force Feedback, Bewegungsplattformen), Olfaktorische Displays (Geruchswahrnehmung, Duftmaschinen)
- 3D Ein-/Ausgabegeräte: 3D-Mäuse, Datenhandschuhe, LeapMotion, optische, magnetische Trackingsysteme (Inside-out, Outside-in)
- 3D Ein-/Ausgabegeräte: Infrarotbasiertes Tracking (Tiefenkamera): Kinect, Gesteninteraktion
- 3D-Displaytechnologien (Stereoskopisch, Volumen/Holografisch), Head-Mounted Displays (HMD), Oculus Rift, Kontaktlinsen als Display
- Ausgabe mittels 3D-Drucker

**Literatur:**

- PREIM, Bernhard, DACHSELT, Raimund, Band 22015. *Interaktive Systeme* [online]. Berlin [u.a.]: Springer PDF e-Book. ISBN 978-3-642-45247-5, 978-3-642-45246-8. Verfügbar unter: <https://doi.org/10.1007/978-3-642-45247-5>.
- PREIM, Bernhard, DACHSELT, Raimund, Band 12010. *Interaktive Systeme* [online]. Berlin [u.a.]: Springer PDF e-Book. ISBN 978-3-642-05401-3, 978-3-642-05402-0. Verfügbar unter: <https://doi.org/10.1007/978-3-642-05402-0>.
- BUTZ, Andreas, KRÜGER, Antonio, 2017. *Mensch-Maschine-Interaktion* [online]. Berlin ; Boston: De Gruyter Oldenbourg PDF e-Book. ISBN 978-3-11-047637-8, 978-3-11-047645-3. Verfügbar unter: <https://doi.org/10.1515/9783110476378>.
- HEINECKE, Andreas M., 2012. *Mensch-Computer-Interaktion: Basiswissen für Entwickler und Gestalter*. 2. Auflage. Heidelberg [u.a.]: Springer. ISBN 978-3-642-13506-4, 978-3-642-13507-1

**Anmerkungen:**

Keine Anmerkungen

<b>Rechnernetze</b>			
<b>Modulkürzel:</b>	IB_RN	<b>SPO-Nr.:</b>	11
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	2
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Sommersemester
<b>Modulverantwortliche(r):</b>	Jarschel, Michael		
<b>Leistungspunkte / SWS:</b>	7 ECTS / 6 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	117 h	
	Selbststudium:	58 h	
	Gesamtaufwand:	175 h	
<b>Lehrveranstaltungen des Moduls:</b>	11.1: Rechnernetze 11.2: Praktikum Rechnernetze		
<b>Lehrformen des Moduls:</b>	11.1: SU/Ü - seminaristischer Unterricht/Übung 11.2: Pr - Praktikum		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
11.1: schrP90 - schriftliche Prüfung, 90 Minuten 11.2: LN - ohne/mit Erfolg teilgenommen  Weitere Erläuterungen: Prüfungsvorleistung: Erfolgreiches Bestehen des integrierten Praktikums mittels Durchführung von mindestens 7 Versuchen. Findet das Praktikum im Online-Betrieb statt, genügt die erfolgreiche Durchführung von 4 Versuchen.			
<b>Voraussetzungen gemäß SPO:</b>			
Keine			
<b>Empfohlene Voraussetzungen:</b>			
Vertrautheit mit heute üblichen netzwerkfähigen Endgeräten (PC, Tablet, DSL-Router, WLAN Access Point, Smart Watch, Alexa, Drucker)			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch der Vorlesung sind die Studierenden in der Lage, <ul style="list-style-type: none"> <li>• Grundlagen der Rechner- und Datenkommunikation, deren Protokolle und Methoden zu verstehen</li> <li>• das Dienste- und Schichten-Konzept des TCP/IP-Architekturmodells und die grundlegende Verteilung der Funktionen zu erklären</li> <li>• die wesentlichen Funktionen der Sicherungsschicht (Framing, Fehlersicherung) zu benennen</li> <li>• die Prinzipien und Parameter zum Mehrfachzugriff in Lokalen Netzen (Ethernet / WLAN) zu beschreiben und Leistungsabschätzungen damit durchzuführen</li> <li>• die Protokoll-Mechanismen für die Sicherung einer Daten-Paket-Übertragung (Sliding Window) zu verstehen</li> <li>• IP-Adressbereiche zu strukturieren und Subnetze zu bilden</li> </ul>			



- die Grundfunktionen eines Routers zu benennen
- die klassischen Routingprotokolle zu erklären und können die Algorithmen zur Erstellung der Routingtabellen anwenden
- die Mechanismen des Transportprotokolls TCP im Detail zu beschreiben und können das Verhalten einer TCP-Verbindung anhand der Systemparameter ingenieurmäßig ableiten und bewerten
- sich selbständig vertiefende Spezialkenntnisse anzueignen, wie auch aufbauende Lehrveranstaltungen zu besuchen

Nach dem Besuch des Praktikums

- haben die Studierenden schwerpunktmäßig die in der Vorlesung erworbenen theoretischen Kenntnisse selbst erprobt und z.T. selbst implementiert.
- haben die Studierenden in einer vorgegebenen Simulationsumgebung für Ausbildungszwecke Module zu Kernfunktionen der Rechnerkommunikation in C++ selbst implementiert und erprobt.
- haben die Studierenden an realen, aktuellen Systemen der Rechnerkommunikation (Schnelle Ethernet-Switche, Router, Hubs) selbst gearbeitet und Aufbau und Funktion einfacher Lokaler Netze selbst durchgeführt.
- beherrschen die Studierenden die Grundfunktionen im Ethernet und den typischen Komponenten dieser Netze.
- haben die Studierenden moderne, komplexe Ethernet-Switche für große Enterprise-Netze kennengelernt, diese selbst administriert und damit größere Netze selbst aufgesetzt. Dabei haben sie sich auch intensiv mit den Protokollen befasst, die für große Ethernet-Netze erforderlich sind (z.B. Spanning Tree, Virtuelle-LAN)
- sind den Studierenden die wesentlichen Elemente einer Datenkommunikation im LAN bekannt und selbst erprobt. Die Konfiguration einfacher Netze sollte jetzt von den Absolventen durchgeführt werden können.

#### Inhalt:

Vorlesung:

- Grundlegendes Verständnis der Rechner- und Datenkommunikation mit Schwerpunkt Paketvermittlung, Architektur-/Schichten-Modell: Protokoll / Dienst
- Grundlagen der Bitübertragung: Grundlagen und Relevante Eigenschaften eines Mediums; Bitrate / Baudrate / Leitungscodierung / Modulationsverfahren
- Grundlagen der Sicherungsschicht: Methoden zur Abgrenzung eines Datenpaketes (Framing); Fehlersicherung mit Zyklischen Codes (CRC); Mehrfachzugriff in Lokalen Netzen; Elemente für Ethernet-Netze (Hub / Switch / VLAN); Wireless LAN / Wi-Fi / Anforderungen und Zugriffsverfahren, Protokoll-Varianten von 802.11; Protokoll-Mechanismen für die Sicherung und Steuerung einer Daten-Paket-Übertragung (Sliding Window)
- Grundlagen der Vermittlungsschicht (IP): IP-Adressen: Struktur und Aufbau, IP-Subnetze, ARP (IPv4); Routing-Grundfunktionen und Struktur eines Routers, Routingprotokolle, Algorithmen dazu; Struktur der IPv6-Adressen
- Grundlagen der Transportschicht (TCP / UDP): Multiplexen in der Transportschicht und NAT
- Eigenschaften UDP
- Eigenschaften TCP und Verhalten einer TCP-Verbindung (Datentransfer und Fehlersicherung, Fluss-Kontrolle, TCP-Stau-Kontrolle, Abschätzen des Durchsatzes einer TCP-Verbindung)

Praktikum:

- Übungen an einer Simulationsumgebung für Protokolle. Dies umfasst die Implementierung und den Test von Basisfunktionen wie z.B. Framing / Bitstuffing, Berechnung des Zyklischen Prüfcodes zur Fehlererkennung (CRC)
- Übungen an und mit realen, aktuellen Systemen der Rechnerkommunikation (Schnelle Ethernet-Switche, Router, Hubs): Aufbau einfacher Lokaler Netze (Hantieren mit unterschiedlichen Netzwerk-Kabeln und Verbindungstypen); Funktionen im Ethernet: (Adresszuordnung (ARP), Funktionen und Leistungsfähigkeit der Vernetzungssysteme Hub und Switch); Administration moderner, komplexer Ethernet-Switche für große Enterprise-Netze; Protokolle und Funktionen für den Betrieb großer Ethernet-Netze:

Spanning Tree und Virtuelle LAN; Einfaches Routing zwischen VLAN und Design und Konfiguration von einem gerouteten Netz mit vollwertigen Routern.
<b>Literatur:</b>
<ul style="list-style-type: none"><li>• SCHREINER, Rüdiger, 2019. <i>Computernetzwerke: von den Grundlagen zur Funktion und Anwendung</i> [online]. München: Hanser PDF e-Book. ISBN 978-3-446-46010-2. Verfügbar unter: <a href="https://doi.org/10.3139/9783446460102">https://doi.org/10.3139/9783446460102</a>.</li><li>• KUROSE, James F. und Keith W ROSS, 2014 . <i>Computernetzwerke : der Top-Down-Ansatz</i> . 6. Auflage. Hallbergmoos: Pearson Studium. ISBN 978-3-86894-237-8 ; 978-3-86326-686-8</li><li>• TANENBAUM, Andrew S. und David WETHERALL, 2012. <i>Computernetzwerke</i>. 5. Auflage. München [u.a.] : Pearson Studium - IT. ISBN 978-3-86894-137-1 ; 3-86894-137-1</li></ul> siehe Vorlesung Rechnernetze
<ul style="list-style-type: none"><li>• Ohne Autor. <i>Empfehlungen der IETF (Spezifikationen zum Internet)</i> [online]. [Zugriff am: 25.09.2023]. Verfügbar unter: <a href="https://www.ietf.org/rfc/">https://www.ietf.org/rfc/</a></li></ul>
<b>Anmerkungen:</b>
Keine Anmerkungen

<b>Algorithmen und Datenstrukturen</b>			
<b>Modulkürzel:</b>	IB_AD	<b>SPO-Nr.:</b>	12
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	3
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Wintersemester
<b>Modulverantwortliche(r):</b>	Glavina, Bernhard		
<b>Leistungspunkte / SWS:</b>	5 ECTS / 4 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	60 h	
	Selbststudium:	48 h	
	Gesamtaufwand:	125 h	
<b>Lehrveranstaltungen des Moduls:</b>	Algorithmen und Datenstrukturen		
<b>Lehrformen des Moduls:</b>	SU/Ü - seminaristischer Unterricht/Übung		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
schrP90 - schriftliche Prüfung, 90 Minuten			
Weitere Erläuterungen: Keine			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Beherrschung mindestens einer imperativen Programmiersprache, sicherer Umgang mit geschachtelten Schleifenkonstrukten, rekursive Programmierung.			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch des Moduls sind die Studierenden in der Lage,			
<ul style="list-style-type: none"> <li>• vorgegebene Algorithmen zu analysieren sowie deren Zeitverhalten und Speicherverbrauch anzugeben.</li> <li>• die wichtigsten Datenstrukturen aufzuzählen sowie deren Vor- und Nachteile zu beschreiben.</li> <li>• zu praktischen Problemstellungen geeignete Datenstrukturen auszuwählen.</li> <li>• Implementierungsaspekte von Algorithmen und Datenstrukturen zu erkennen und deren Auswirkungen zu beschreiben.</li> <li>• die Algorithmen in einer imperativen oder objektorientierten Programmiersprache umzusetzen.</li> <li>• Laufzeitmessungen für die implementierten Algorithmen durchzuführen und mit den Mitteln der deskriptiven Statistik auszuwerten und darzustellen.</li> </ul>			
<b>Inhalt:</b>			
<ul style="list-style-type: none"> <li>• Komplexität und Effizienz</li> <li>• Listen und Reihungen</li> </ul>			

- Graphen und Relationen
- Bäume und Hierarchien
- Streuspeicherverfahren
- Sortierverfahren
- Ausgewählte Anwendungen

**Literatur:**

- SAAKE, Hunter und Kai-Uwe SATTLER, 2021. *Algorithmen und Datenstrukturen : eine Einführung mit Java* . 6. Auflage. ISBN 978-3-86490-769-2
- BRUCE, Peter, Andrew BRUCE und Peter GEDECK, 2021. *Praktische Statistik für Data scientists : 50+ essenzielle Konzepte mit R und Python*. ISBN 978-3-96009-153-0
- KNEBL, Helmut, 2021. *Algorithmen und Datenstrukturen: Grundlagen und probabilistische Methoden für den Entwurf und die Analyse* [online]. Wiesbaden: Springer Vieweg PDF e-Book. ISBN 978-3-658-32714-9. Verfügbar unter: <https://doi.org/10.1007/978-3-658-32714-9>.

**Anmerkungen:**

Keine Anmerkungen

<b>Software Engineering</b>			
<b>Modulkürzel:</b>	IB_SE	<b>SPO-Nr.:</b>	13
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	3
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Wintersemester
<b>Modulverantwortliche(r):</b>	Kugele, Stefan		
<b>Leistungspunkte / SWS:</b>	7 ECTS / 6 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	70 h	
	Selbststudium:	105 h	
	Gesamtaufwand:	175 h	
<b>Lehrveranstaltungen des Moduls:</b>	13.1: Software Engineering 13.2: Praktikum Software Engineering		
<b>Lehrformen des Moduls:</b>	13.1: SU/Ü - seminaristischer Unterricht/Übung 13.2: Pr - Praktikum		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
13.1: schrP90 - schriftliche Prüfung, 90 Minuten 13.2: LN - ohne/mit Erfolg teilgenommen  Weitere Erläuterungen: Praktikum: Zum erfolgreichen Bestehen des Praktikums ist die Lösung und Dokumentation aller 7 Praktikumsaufgaben erforderlich. Die genauen Abgabetermine werden zu Beginn des Semesters durch den Dozenten bekannt gegeben.			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Beherrschung der objektorientierten Programmiersprache Java			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch der Vorlesung sind die Studierenden in der Lage, <ul style="list-style-type: none"> <li>• kleinere und mittlere Softwaresysteme zu entwickeln</li> <li>• die grundlegenden Schritte des Software-Engineering zu benennen</li> <li>• existierende Qualitätsmodelle und deren Bedeutung für die Entwicklung von Software zu benennen</li> <li>• Anforderungen an ein Softwaresystem strukturiert zu beschreiben</li> <li>• grundlegende Architekturprinzipien zur Erstellung eigener SW-Architekturen anzuwenden</li> <li>• ausgewählte Diagramme der UML zur Beschreibung und Dokumentation einer Software einzusetzen</li> <li>• verschiedene Teststrategien auf eigene Problemstellungen anzuwenden</li> <li>• grundlegende Vorgehensmodelle für die Software Entwicklung zu benennen und anzuwenden</li> </ul>			

<p><b>Praktikum:</b></p> <ul style="list-style-type: none"> <li>• die in der Vorlesung behandelten Methoden zur Entwicklung von Software-Systemen praktisch einzusetzen</li> <li>• Softwaresysteme zu analysieren, zu planen und umzusetzen</li> <li>• Anforderungen an ein Softwareprodukt strukturiert zu dokumentieren</li> <li>• Softwaresysteme mit Hilfe von UML-Diagrammen zu beschreiben</li> <li>• Software-Architekturen zu entwerfen und zu dokumentieren</li> <li>• Testfälle zu spezifizieren und Testdurchführungen zu dokumentieren</li> </ul>
<p><b>Inhalt:</b></p> <p>Vorlesung:</p> <ul style="list-style-type: none"> <li>• Grundlagen: Software Engineering, Software Qualität</li> <li>• Requirements Engineering: Vorgehensweise, Stakeholder, Systemkontext, Erhebungsmethoden, UML-Diagramme</li> <li>• Software Architektur: Grundlagen, Architekturprinzipien, Komponentenarchitektur, Entity-Boundary-Controller, Design &amp; Implementierung, Umsetzung von Klassendiagrammen und Zustandsautomaten, Coding-Rules</li> <li>• Software Architektur: Dokumentation</li> <li>• Test: Grundlagen</li> <li>• Test: Blackboxtesting</li> <li>• Test: Whiteboxtesting</li> </ul> <p>Praktikum:</p> <p>Bearbeiten von mehreren Praktikumsaufgaben mit folgenden Themenschwerpunkten:</p> <ul style="list-style-type: none"> <li>• Requirements Engineering: Stakeholder, Systemkontext, Use-Cases, Klassendiagramme, Zustandsdiagramme</li> <li>• Requirements Engineering: Dokumentation</li> <li>• Software Architektur &amp; Design: Komponentenarchitektur, Entity-Boundary-Controller, Sequenz-Diagramme, Komponentendiagramme</li> <li>• Implementierung: Umsetzung einer Komponentenarchitektur</li> <li>• Test: Blackboxtesting</li> <li>• Test: Whiteboxtesting</li> </ul>
<p><b>Literatur:</b></p> <ul style="list-style-type: none"> <li>• Ian Sommerville, „Software Engineering“, Pearson Studium</li> <li>• Chris Rupp, Stefan Queins 6 die sophisten: "UML 2 glasklar: Praxiswissen für die UML-Modellierung", Carl Hanser Verlag GmbH &amp; Co. KG</li> <li>• Andreas Spillner, Tilo Linz: "Basiswissen Softwaretest: Aus- und Weiterbildung zum Certified Tester - Foundation Level nach ISTQB-Standard (ISQL-Reihe)", dpunkt.verlag GmbH</li> <li>• UML 2 für Studenten; Harald Störrle; Pearson Studium; 2005</li> <li>• SOMMERVILLE, Ian, 2018. <i>Software Engineering</i>. 10. Auflage. Hallbergmoos: Pearson. ISBN 978-3-86894-344-3, 3-86894-344-7</li> <li>• SPILLNER, Andreas und Tilo LINZ, 2019. <i>Basiswissen Softwaretest: Aus- und Weiterbildung zum Certified Tester : Foundation Level nach ISTQB-Standard</i>. 6. Auflage. Heidelberg: dpunkt.verlag. ISBN 978-3-86490-583-4, 3-86490-583-4</li> <li>• STÖRRLE, Harald, 2005. <i>UML 2 für Studenten: [mit UML-Syntax-Poster]</i>. München [u.a.]: Pearson. ISBN 3-8273-7143-0</li> </ul>
<p><b>Anmerkungen:</b></p> <p>Keine Anmerkungen</p>

<b>Webtechnologien</b>			
<b>Modulkürzel:</b>	IB_WEB	<b>SPO-Nr.:</b>	14
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	3
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Wintersemester
<b>Modulverantwortliche(r):</b>	Windisch, Hans-Michael		
<b>Leistungspunkte / SWS:</b>	7 ECTS / 6 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	70 h	
	Selbststudium:	105 h	
	Gesamtaufwand:	175 h	
<b>Lehrveranstaltungen des Moduls:</b>	14.1: Webtechnologien 14.2: Praktikum Webtechnologien		
<b>Lehrformen des Moduls:</b>	14.1: SU/Ü - seminaristischer Unterricht/Übung 14.2: Pr - Praktikum		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
14.1: schrP90 - schriftliche Prüfung, 90 Minuten 14.2: LN - ohne/mit Erfolg teilgenommen  Weitere Erläuterungen: Zum erfolgreichen Bestehen des Praktikums ist die Lösung von 4 Praktikumsaufgaben erforderlich.			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Beherrschung der Programmiersprache Java			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch der Lehrveranstaltung sind die Studierenden in der Lage, <ul style="list-style-type: none"> <li>• die grundlegenden Begriffe und Funktionsweisen des World Wide Web (WWW) zu benennen</li> <li>• ihre theoretischen Kenntnisse über moderne Client- und serverseitige Webtechnologien strukturiert zu beschreiben</li> <li>• Webseiten mit HTML zu spezifizieren und durch den Einsatz von CSS das Layout der Seiten zu gestalten</li> <li>• dynamische Webseiten zu entwickeln, indem sie auf die entsprechenden Möglichkeiten von JavaScript zurückgreifen</li> <li>• wesentliche Möglichkeiten von Bootstrap zu benennen und diese zur Gestaltung ihrer Webseiten einzusetzen.</li> <li>• das Backend von Webanwendungen mit PHP zu entwickeln</li> <li>• Webservices und REST zur Definition von Server-Schnittstellen einzusetzen und Server mit diesen Schnittstellen zu entwickeln</li> </ul>			

<p>Praktikum:</p> <ul style="list-style-type: none"><li>• Werkzeuge für GUI-Prototyping mit HTML und CSS anzuwenden</li><li>• die GUI mit Hilfe von JavaScript zu implementieren</li><li>• ein Backend mit PHP zum implementieren</li></ul>
<p><b>Inhalt:</b></p> <ul style="list-style-type: none"><li>• Grundlegende Kenntnisse des WWW</li><li>• Client-seitige Technologien: Hypertext Markup Language (HTML5)</li><li>• Client-seitige Technologien: Cascading Stylesheets (CSS3)</li><li>• Client-seitige Technologien: JavaScript, Domain Object Model (DOM)</li><li>• Client-seitige Technologien: Bootstrap als CSS-Framework</li><li>• Server-seitige Technologien: PHP</li><li>• Server-seitige Technologien: Sessions und Cookies</li><li>• Server-seitige Technologien: Asynchronous JavaScript and XML (AJAX)</li><li>• Server-seitige Technologien: Representational State Transfer (REST)</li></ul> <p>Praktikum:</p> <ul style="list-style-type: none"><li>• GUI-Prototyping mit HTML und CSS</li><li>• Webclient-Programmierung mit JavaScript</li><li>• PHP-Programmierung</li></ul>
<p><b>Literatur:</b></p> <p>Wird im Verlauf der Veranstaltung zu den einzelnen Abschnitten in Form von Weblinks bekannt gegeben.</p> <ul style="list-style-type: none"><li>• FLANAGAN, David, 2020. <i>JavaScript - The Definitive Guide</i>. 7. Auflage. ISBN 978-1491952023</li><li>• WOLF, Jürgen, 2019. <i>HTML5 und CSS3 : das umfassende Handbuch</i>. 3. Auflage. Bonn: Rheinwerk Computing. ISBN 978-3-8362-4158-8 ; 3-8362-4158-7</li><li>• HELLER, Stephan, . <i>PHP 5.6 - Grundlagen zur Erstellung dynamischer Webseiten</i>. Bodenheim: Herdt. ISBN s. <a href="https://herdt-campus.de/product/GPHP56">https://herdt-campus.de/product/GPHP56</a></li></ul>
<p><b>Anmerkungen:</b></p> <p>Keine Anmerkungen</p>



<b>Verteilte Systeme</b>			
<b>Modulkürzel:</b>	IB_VTS	<b>SPO-Nr.:</b>	15
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	3
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Wintersemester
<b>Modulverantwortliche(r):</b>	Facchi, Christian		
<b>Leistungspunkte / SWS:</b>	6 ECTS / 5 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	59 h	
	Selbststudium:	91 h	
	Gesamtaufwand:	150 h	
<b>Lehrveranstaltungen des Moduls:</b>	15.1: Verteilte Systeme 15.2: Praktikum Verteilte Systeme		
<b>Lehrformen des Moduls:</b>	15.1: SU/Ü - seminaristischer Unterricht/Übung 15.2: Pr - Praktikum		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
15.1: schrP90 - schriftliche Prüfung, 90 Minuten 15.2: LN - ohne/mit Erfolg teilgenommen  Weitere Erläuterungen: Als Leistungsnachweis ist die erfolgreiche Bearbeitung von 5 Übungsblättern erforderlich. Hierbei werden folgende Kenntnisse überprüft: <ul style="list-style-type: none"> <li>• Fähigkeit zur praktischen Problemlösung in verteilten Systemen</li> <li>• Verwendung von Semaphoren und Monitore</li> <li>• Verwendung von C und Java zur Bewältigung von Problemen aus dem Bereich der parallelen Programmierung und verteilte Systeme (Dining Philosophers, Reader-Writer Problem, RPC, RMI)</li> </ul>			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Beherrschung der Programmiersprache C, insbesondere Records, mehrdimensionale Arrays, Pointer. Beherrschung der Programmiersprache Java, insbesondere Threads.			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch des Moduls sind die Studierenden in der Lage, <ul style="list-style-type: none"> <li>• ihre Kenntnisse über grundlegende Paradigmen für verteilte Systeme in der Praxis einzusetzen</li> <li>• ihre Kenntnisse über grundlegende Probleme im Bereich verteilter Systeme und insbesondere der parallelen Programmierung und deren Beseitigung zu verwenden</li> <li>• Sprachmittel der Programmiersprachen C und JAVA zur Beschreibung von verteilten Systemen praxisnah einzusetzen</li> </ul>			

<ul style="list-style-type: none"><li>• ihre Kenntnisse zum Einsatz von Spezifikationssprachen (z.B.: UML, Prozesskalküle) und zur Beschreibung von verteilten Systemen sowohl abstrakt als auch am konkreten Beispiel einzusetzen</li><li>• ihre Kenntnisse zur Analyse und zum Design von verteilten Systemen auf konkrete Problemstellungen anzuwenden</li><li>• das Problem des gegenseitigen Ausschlusses im Kontext bekannter Standardanwendungsfälle (Barriersynchronisation, Read/Writer, Consumer/Producer) in den Sprachen C und Java zu lösen</li></ul>
<b>Inhalt:</b>
<p>Vorlesung:</p> <ul style="list-style-type: none"><li>• Wiederholung Parallele Programmierung</li><li>• Grundlagen Verteilter Systeme</li><li>• Modellierung Verteilter Systeme (Prozesskalküle)</li><li>• Kommunikation von Systemen</li><li>• Synchronisation von Systemen</li><li>• Sprachmittel zur Beschreibung von parallelen und kommunizierenden Programmen</li><li>• Probleme bei Verteilten Systemen (Safety, Security, Abstimmungsverfahren, ...)</li><li>• Beschreibung von Verteilten Systemen (UML, Statecharts, ...)</li><li>• Beispiele von Verteilten Systemen</li></ul> <p>Praktikum:</p> <ul style="list-style-type: none"><li>• Synchronisation von Verteilten Systemen in C und Java</li><li>• RPC</li><li>• RMI</li></ul>
<b>Literatur:</b>
<ul style="list-style-type: none"><li>• TANENBAUM, Andrew und Marten VAN STEEN, 2007. <i>Distributed Systems</i>. 2. Auflage.</li><li>• COULOURIS, George F., Jean DOLLIMORE und Tim KINDBERG, 2012. <i>Distributed systems: concepts and design</i>. 5. Auflage. Boston [u.a.]: Addison-Wesley. ISBN 978-0-273-76059-7</li><li>• SCHILL, Alexander und Thomas SPRINGER, 2012. <i>Verteilte Systeme</i>.</li></ul>
<b>Anmerkungen:</b>
Keine Anmerkungen

<b>Wissensbasierte Systeme</b>			
<b>Modulkürzel:</b>	IB_WBS	<b>SPO-Nr.:</b>	16
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	3
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Wintersemester
<b>Modulverantwortliche(r):</b>	Hahndel, Stefan		
<b>Leistungspunkte / SWS:</b>	5 ECTS / 4 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	47 h	
	Selbststudium:	78 h	
	Gesamtaufwand:	125 h	
<b>Lehrveranstaltungen des Moduls:</b>	Wissensbasierte Systeme		
<b>Lehrformen des Moduls:</b>	SU/Ü - seminaristischer Unterricht/Übung		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
schrP90 - schriftliche Prüfung, 90 Minuten			
Weitere Erläuterungen: Keine			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Grundkenntnisse in Aussagen- und Prädikatenlogik (Lesen von Formeln), Spezifizieren und Modellieren einfacher Sachverhalte mit Hilfe von Formeln			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch des Moduls sind die Studierenden in der Lage,			
<ul style="list-style-type: none"> <li>• den Nutzen und die Einsatzmöglichkeiten Wissensbasierter Systeme zu erkennen und zu verstehen, wie dieses Gebiet mit anderen Kernbereichen der Informatik zusammenhängt bzw. von diesen abzugrenzen ist</li> <li>• die klassischen Verfahren zur Darstellung, Erhebung und zur maschinellen Verarbeitung menschlichen Wissens als Grundlage der Künstlichen Intelligenz zu benennen</li> <li>• diese klassischen Verfahren anhand einer gängigen KI-Programmiersprache praktisch einzusetzen. Dabei stehen wirtschaftlich verwertbare Technologien im Vordergrund, wie sie Expertensysteme und lernende Klassifikatoren beinhalten</li> <li>• ausgewählte logische Probleme unter Einsatz einer logischen Programmiersprache methodisch zu lösen</li> <li>• ausgewählte topologische Spielprobleme methodisch ohne Rechnerunterstützung zu lösen, falls diese an Handexemplaren untersucht werden können</li> <li>• ausgewählte topologische Spielprobleme unter Einsatz einer logischen Programmiersprache oder eines SAT-Beweislers zu modellieren und zu lösen</li> </ul>			

<b>Inhalt:</b>
<ul style="list-style-type: none"><li>• Grundlagen der Künstlichen Intelligenz (KI)</li><li>• Logikprogrammierung (Prolog)</li><li>• Problemlösen mit Graphen: Anwendung auf Spielprobleme</li><li>• Constraint Satisfaction Problems und Constraint Logic Programming</li><li>• Wissensrepräsentation: Logik, Inferenz, Regelbasierte und Expertensysteme</li><li>• Maschinelles Lernen (Beispiel Decision Trees)</li><li>• Grundlagen der Mustererkennung (Bayes, Klassifikation, Loss Matrix, DTW)</li><li>• Anwendung: Dynamische Gestenerkennung</li></ul>
<b>Literatur:</b>
<ul style="list-style-type: none"><li>• BEIERLE, Christoph, KERN-ISBERNER, Gabriele, 2019. <i>Methoden wissensbasierter Systeme: Grundlagen, Algorithmen, Anwendungen</i> [online]. Wiesbaden: Springer Vieweg PDF e-Book. ISBN 978-3-658-27084-1. Verfügbar unter: <a href="https://doi.org/10.1007/978-3-658-27084-1">https://doi.org/10.1007/978-3-658-27084-1</a>.</li><li>• CLOCKSIN, William F. und Christopher S. MELLISH, 2003. <i>Programming in Prolog: [using the ISO standard]</i>. 5. Auflage. Berlin [u.a.]: Springer. ISBN 3-540-00678-8, 0-387-00678-8</li><li>• RUSSELL, Stuart J. und Peter NORVIG, 2012. <i>Künstliche Intelligenz: ein moderner Ansatz</i>. 3. Auflage. München ; Harlow ; Amsterdam ; Madrid ; Boston ; San Francisco ; Don Mills ; Mexico City ; Sydney: Pearson. ISBN 978-3-86894-098-5, 3-86894-098-7</li></ul>
<b>Anmerkungen:</b>
Keine Anmerkungen

<b>Datenbanksysteme</b>			
<b>Modulkürzel:</b>	IB_DBS	<b>SPO-Nr.:</b>	17
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	4
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Sommersemester
<b>Modulverantwortliche(r):</b>	Navarro Bullock, Beate		
<b>Leistungspunkte / SWS:</b>	7 ECTS / 6 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	70 h	
	Selbststudium:	105 h	
	Gesamtaufwand:	175 h	
<b>Lehrveranstaltungen des Moduls:</b>	17.1: Datenbanksysteme 17.2: Praktikum Datenbanksysteme		
<b>Lehrformen des Moduls:</b>	17.1: SU/Ü - seminaristischer Unterricht/Übung 17.2: Pr - Praktikum		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
17.1: schrP90 - schriftliche Prüfung, 90 Minuten 17.2: LN - ohne/mit Erfolg teilgenommen  Weitere Erläuterungen: Zum Bestehen des Praktikums müssen insgesamt vier Testate erworben werden. Dazu gibt es im Laufe des Semesters vier Aufgabenstellungen bei deren erfolgreicher Bearbeitung jeweils ein Testat vergeben wird. Die fertigen Lösungen sind fristgerecht einzureichen oder in einem Präsenztermin zu präsentieren, wobei auch Fragen zum Lösungskonzept zu beantworten sind.			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Kenntnisse mindestens einer höheren Programmiersprache, die Schnittstellen für die Nutzung von SQL anbietet			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch der Vorlesung sind die Studierenden in der Lage, <ul style="list-style-type: none"> <li>• die grundlegenden Prinzipien und Konzepte (relationaler) Datenbanksysteme zu benennen und können diese als zentrale fachliche und technologische Infrastruktur-Komponenten zur Datenhaltung in den Kontext unternehmensspezifischer Informationssysteme einordnen</li> <li>• die Grundlagen der Datenbankarchitektur, des Datenbankentwurfs und Datenbankabfragen zu beschreiben und die wichtigsten hiermit verbundenen Konzepte und Abstraktionsmechanismen zu beschreiben</li> <li>• abzuwägen, ob und wie diese zur Umsetzung konkreter fachlicher Anforderungen genutzt werden können</li> </ul>			

<ul style="list-style-type: none"> <li>• (Datenbank-) Schemata zu erstellen</li> <li>• Anfrage- bzw. Änderungsoperationen in der relationalen Algebra und SQL zu formulieren</li> <li>• die Entwicklung von Datenbanksystem von relational bis NoSQL nachzuvollziehen</li> </ul> <p>Praktikum:</p> <ul style="list-style-type: none"> <li>• die grundlegenden Prinzipien und Konzepte des konzeptionellen und systemnahen Datenbankentwurfs zur Umsetzung konkreter fachlicher Anforderungen anzuwenden, ein für die Anforderungen geeignetes Datenbankschema zu erstellen und in einem relationalen Datenbanksystem zu implementieren</li> <li>• zu fachlich formulierten Datenabfragen und –änderungen korrespondierende SQL-Anfragen zu formulieren und diese an ein Datenbanksystem zu stellen</li> <li>• die Auswirkungen verschiedener Freiheitsgrade bei Entwurf, Modellierung und Abfrage von Datenbanken zu benennen und können diese beim Erstellen von Schemata bzw. Anfragen berücksichtigen</li> </ul>
<p><b>Inhalt:</b></p> <p>Vorlesung:</p> <ul style="list-style-type: none"> <li>• Grundlagen von Datenbanksystemen: Historie, Konzepte und Architektur; 3-Schichten-Modell und Datenunabhängigkeit</li> <li>• Konzeptioneller (fachlicher) Datenbankentwurf und Entity-Relationship-Modell</li> <li>• Datenintegrität und Integritätsbedingungen</li> <li>• Relationales Datenmodell und Relationenalgebra</li> <li>• Relationaler Datenbankentwurf und Normalformen</li> <li>• SQL</li> <li>• Transaktionen und Transaktionsmanagement</li> <li>• Physische Datenorganisation</li> <li>• Weiterführende Datenbank-Konzepte und thematischer Ausblick</li> </ul> <p>Praktikum:</p> <ul style="list-style-type: none"> <li>• Konzeptionelle Modellierung und Entwurf von Datenbankschemata</li> <li>• Schemaimplementierung</li> <li>• SQL: Data Definition Language (DDL)</li> <li>• SQL: Data Manipulation Language (DML) und Datenbankabfragen</li> <li>• Umgang mit unvollständiger Information und Nullwerten</li> <li>• Weiterführende Konzepte</li> </ul>
<p><b>Literatur:</b></p> <ul style="list-style-type: none"> <li>• ELMASRI, Ramez A. und Shamkant B. NAVATHE, 2009. <i>Grundlagen von Datenbanksystemen</i>. 3. Auflage.</li> <li>• UNTERSTEIN, Michael und Günter MATTHIESSEN, 2012. <i>Relationale Datenbanken und SQL in Theorie und Praxis</i>. 5. Auflage.</li> <li>• KEMPER, Alfons und André EICKLER, 2013. <i>Datenbanksysteme: Eine Einführung</i>. 9. Auflage.</li> <li>• VOSSSEN, Gottfried, 2008. <i>Datenmodelle, Datenbanksprachen und Datenbankmanagementsysteme</i>. 5. Auflage.</li> </ul>
<p><b>Anmerkungen:</b></p> <p>Keine Anmerkungen</p>

<b>Fachwissenschaftliches Seminar</b>			
<b>Modulkürzel:</b>	IB_SEM	<b>SPO-Nr.:</b>	18
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	4
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Sommersemester
<b>Modulverantwortliche(r):</b>	Gold, Robert		
<b>Leistungspunkte / SWS:</b>	3 ECTS / 2 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	24 h	
	Selbststudium:	51 h	
	Gesamtaufwand:	75 h	
<b>Lehrveranstaltungen des Moduls:</b>	Fachwissenschaftliches Seminar		
<b>Lehrformen des Moduls:</b>	S - Seminar		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
SA - Seminararbeit mit Präsentation			
Weitere Erläuterungen:			
<ul style="list-style-type: none"> <li>Die Prüfungsleistung in diesem Modul ist eine Seminararbeit.</li> <li>Laut SPO ist die Seminararbeit eine Hausarbeit mit mündlicher Präsentation. Umfang der Hausarbeit lt. APO: 3000 bis 6000 Wörter, ca. 10 bis 20 Seiten. Die Seminararbeit (Hausarbeit + Präsentation) ist mit einer Textverarbeitungssoftware zu erstellen.</li> <li>Den Teilnehmern wird zu Beginn des Seminars ein Thema per Los oder Wahl zugeteilt</li> <li>In der Vorbereitungsphase muss jeder Teilnehmer Literaturrecherchen zu seinem Thema durchführen und deren Ergebnis in seine Präsentation und seine schriftliche Ausarbeitung einfließen lassen</li> <li>Im Zuge des Seminars muss jeder Teilnehmer eine ganze Doppelstunde (90 Minuten) über ein Thema gestalten. Im Rahmen der Doppelstunde hält der Teilnehmer einen Vortrag (ca. 45 Minuten), zeigt gegebenenfalls Demonstrationen und moderiert eine fachliche Diskussion im Nachgang des Vortrags</li> <li>In diesem Modul besteht Anwesenheitspflicht</li> </ul>			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Sicherer Umgang mit einem Werkzeug zur Erstellung von Präsentationen. Vorerfahrung in Vortragsgestaltung und Präsentation			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch des Moduls sind in Studierenden in der Lage, <ul style="list-style-type: none"> <li>sich selbständig spezielle fachliche Kenntnisse zu erarbeiten (Literaturarbeit, Analyse, Schlussfolgerungen) und können diese mithilfe des Einsatzes geeigneter Medien nachvollziehbar im Rahmen eines mündlichen Vortrags präsentieren</li> </ul>			

<ul style="list-style-type: none"><li>• abhängig vom Zuhörerkreis ihren Vortrag durch Demonstrationen am Rechner anzureichern</li><li>• abhängig vom Zuhörerkreis ihren Vortrag durch technische Details anzureichern, um so während oder nach dem Vortrag eine fachliche Diskussion zu motivieren</li><li>• einer fachlichen Präsentation kritisch zu folgen und die Inhalte mit dem Vortragenden fachlich zu diskutieren (Stärkung der kommunikativen Kompetenz)</li><li>• im Nachgang einer Präsentation dem Vortragenden hilfreiches und faires Feedback zu liefern</li><li>• ihre überfachlichen und kommunikativen Kompetenzen für die Gestaltung und Präsentation eines Fachvortrags zu nutzen</li><li>• den Inhalt ihrer Präsentation in Form einer kurzen schriftlichen Ausarbeitung darzustellen</li></ul>
<b>Inhalt:</b>
<ul style="list-style-type: none"><li>• Lesen wissenschaftlicher Fachliteratur und die Erschließung fachlicher Inhalte</li><li>• Gestaltung von Online-Demonstrationen</li><li>• Darstellung und Diskussion von Quell-Code, Software-Architekturen und Software-Designs in Fachvorträgen</li><li>• Das Führen fachlicher Diskussionen</li><li>• Fachvorträge aktiv mitverfolgen und fachliche Diskussionsbeiträge liefern</li><li>• Faires Feedback geben</li><li>• Verständlich technisch und nicht-technisch Kommunizieren</li><li>• Fachliche Aspekte eines Vortrags strukturiert dokumentieren</li></ul>
<b>Literatur:</b>
<ul style="list-style-type: none"><li>• VOSS, Rödiger, 2022. <i>Wissenschaftliches Arbeiten: ... leicht verständlich : mit zahlreichen Abbildungen und Übersichten</i> [online]. München: UVK PDF e-Book. ISBN 978-3-8385-8812-4. Verfügbar unter: <a href="https://elibrary.utb.de/doi/book/10.36198/9783838588124">https://elibrary.utb.de/doi/book/10.36198/9783838588124</a>.</li><li>• RESTLE, Viola, 2022. <i>30 Minuten Zahlen lebendig präsentieren</i>. 1. Auflage. Offenbach: Gabal Verlag GmbH. ISBN 978-3-96740-183-7</li><li>• BEUSHAUSEN, Ulla, 2023. <i>Sicher und frei reden: Sprechängste erfolgreich abbauen - Trainingsprogramm mit 6 Bausteinen</i>. 5. Auflage. München: Ernst Reinhardt Verlag. ISBN 978-3-497-03206-8</li></ul>
<b>Anmerkungen:</b>
<ul style="list-style-type: none"><li>• Das fachliche Thema des Seminars wechselt von Kurs zu Kurs. Gegenstand ist zumeist ein studien-gangspezifisches Gebiet, zu dem es geeignete Fachliteratur gibt, die zugleich die Basisliteratur für die Vorträge darstellt. Die Literatur wird vom jeweiligen Dozenten zur Verfügung gestellt.</li><li>• Detaillierte Hinweise zu Terminen und seine Erwartungen hinsichtlich der Gestaltung der Präsentation sowie der schriftlichen Ausarbeitung kommuniziert der jeweilige Dozent zu Beginn des Semesters.</li></ul>



<b>Grundlagen der Theoretischen Informatik</b>			
<b>Modulkürzel:</b>	IB_GTHI	<b>SPO-Nr.:</b>	19
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	4
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Sommersemester
<b>Modulverantwortliche(r):</b>	Regensburger, Franz		
<b>Leistungspunkte / SWS:</b>	5 ECTS / 4 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	47 h	
	Selbststudium:	78 h	
	Gesamtaufwand:	125 h	
<b>Lehrveranstaltungen des Moduls:</b>	Grundlagen der Theoretischen Informatik		
<b>Lehrformen des Moduls:</b>	SU/Ü - seminaristischer Unterricht/Übung		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
schrP90 - schriftliche Prüfung, 90 Minuten			
Weitere Erläuterungen: Keine			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Vertrautheit mit der mathematischen Formelsprache, Grundlagen der Mengenlehre (Funktionen, Relationen)			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch des Moduls sind die Studierenden in der Lage,			
<ul style="list-style-type: none"> <li>• grundlegende Konzepte, Begriffe und Zusammenhänge der Teilgebiete Formale Sprachen, Automatentheorie, Berechenbarkeits- und Komplexitätstheorie zu erläutern</li> <li>• einfache Beweise selbst zu führen und dabei ihre Problemlösungskompetenz einzusetzen</li> <li>• die Leistungsfähigkeit der Beschreibungsmittel formaler Sprachen zu differenzieren (Chomsky-Hierarchie)</li> <li>• formale Sprachen mit den dargestellten Mitteln selbst zu beschreiben</li> <li>• wichtige abstrakte Berechnungsmodelle zu beschreiben und wissen, dass es Grenzen der Berechenbarkeit gibt</li> <li>• wichtige historische Meilensteine der Informatik zu benennen sowie deren Einfluss auf die heutige und zukünftige Entwicklung zu erkennen</li> </ul>			

<b>Inhalt:</b>
<ul style="list-style-type: none"><li>• Grundbegriffe: Alphabete, Wörter, Formale Sprachen, Grammatiken, Chomsky-Hierarchie, Wortproblem</li><li>• Reguläre Sprachen: deterministische und nichtdeterministische endliche Automaten, Potenzmengenkonstruktion, reguläre Ausdrücke, Minimierung, Pumping-Lemma, Satz von Myhill-Nerode, Abschlusseigenschaften</li><li>• Kontextfreie Sprachen: Normalformen, Pumping-Lemma, Abschlusseigenschaften, CYK-Algorithmus, Kellerautomaten</li><li>• Kontextsensitive und Typ-0-Sprachen: linear beschränkte Automaten, Turing-Maschinen</li><li>• Grundbegriffe: Abzählbarkeit, Berechenbarkeit</li><li>• Abstrakte Rechner- und Programmiermodelle: Turingmaschinen, Abacus-Maschinen</li><li>• Entscheidungsprobleme: Halteproblem, Reduktionsprinzip, Satz von Rice</li></ul>
<b>Literatur:</b>
<ul style="list-style-type: none"><li>• ASTEROTH, Alexander und Christel BAIER, 2003. <i>Theoretische Informatik: eine Einführung in Berechenbarkeit, Komplexität und formale Sprachen mit 101 Beispielen</i>. München: Pearson Studium. ISBN 978-3-8632-6664-6, Print 3-8273-7033-7</li><li>• BOOLOS, BURGESS und JEFFREY, 2007. <i>Computability and Logic</i>. ISBN 978-0-521-70146-4</li></ul>
<b>Anmerkungen:</b>
Keine Anmerkungen

<b>Diskrete Mathematik</b>			
<b>Modulkürzel:</b>	IB_DM	<b>SPO-Nr.:</b>	20
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	4
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Sommersemester
<b>Modulverantwortliche(r):</b>	Krüger, Max		
<b>Leistungspunkte / SWS:</b>	5 ECTS / 4 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	47 h	
	Selbststudium:	78 h	
	Gesamtaufwand:	125 h	
<b>Lehrveranstaltungen des Moduls:</b>	Diskrete Mathematik		
<b>Lehrformen des Moduls:</b>	SU/Ü - seminaristischer Unterricht/Übung		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
schrP90 - schriftliche Prüfung, 90 Minuten			
Weitere Erläuterungen: Keine			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Mathematische Grundlagen 1 und 2			
<b>Angestrebte Lernergebnisse:</b>			
Dieses Modul dient ...			
<ul style="list-style-type: none"> <li>• dem Ausbau des Verständnisses der Grundlagen aus dem Bereich der diskreten Mathematik,</li> <li>• Vorbereitung auf weiterführende Fächer im Bereich der angewandten und theoretischen Informatik und</li> <li>• der Bereitstellung (diskreter) algorithmischer Verfahren für Anwendungen inner- und außerhalb der Informatik</li> </ul>			
anhand von ausgewählten Themen aus der diskreten Mathematik.			
Nach dem Besuch des Moduls sind die Studierenden in der Lage,			
<ul style="list-style-type: none"> <li>• grundlegende mathematische Begriffe, deren Eigenschaften und Zusammenhänge sowie darauf aufbauende Algorithmen zu benennen</li> <li>• die Begriffe, Zusammenhänge und Algorithmen am Beispiel zu erläutern und deren wesentliche Funktionsweisen zu verstehen</li> </ul>			

- die Bedeutung und den Nutzen mathematischer Formelnotation bei der Beschreibung und Behandlung von Anwendungsproblemen zu erkennen
- eigenständig typische Aufgabenstellungen zu lösen
- bei der Bearbeitung von Anwendungsproblemen auftretende, grundlegende mathematische Problemstellungen zu erkennen und diese mit geeigneten Verfahren zu lösen
- einfache mathematische Beweise zu führen
- Ergebnisse kritisch hinsichtlich ihrer mathematischen Korrektheit zu hinterfragen
- Ergebnisse kritisch hinsichtlich ihrer Aussage für die zugrunde liegenden Anwendungsprobleme zu prüfen und zu beurteilen

**Inhalt:**

In der Veranstaltung wird eine Auswahl aus den folgenden Themengebieten der diskreten Mathematik behandelt:

- Graphentheorie: Ungerichtete und gerichtete Graphen; Grundlegende Graphenalgorithmien; Graphentheoretische Beweise; Wege und Kreise; bipartite Graphen; Anwendungen der Graphentheorie
- Elementare Zahlentheorie: natürliche, ganze und Primzahlen; Teilbarkeit und Kongruenz; chinesischer Restsatz; Euklidischer Algorithmus; quadratische Reziprozitätsgesetz
- Ordnungs- und Verbandstheorie: Relationen, Ordnungen und Halbordnungen; Äquivalenzrelationen, Äquivalenzklassen und Quotientenbildung; Fixpunktsatz von Knaster-Tarski
- Lineare Optimierung: Modellierung
- Lineare Optimierung mit dem Simplex-Algorithmus
- Lineare Optimierung: Binäre und ganzzahlige lineare Optimierung
- Exemplarisches Programmpaket zur linearen Optimierung; Anwendungen der linearen Optimierung

**Literatur:**

- IHRINGER, T., 2002. *Diskrete Mathematik*.
- HUßMANN, Peter und Brigitte LUTZ-WESTPHAL, 2015. *Diskrete Mathematik erleben : Anwendungs-basierte und verstehensorientierte Zugänge*. 2. Auflage. Wiesbaden: Springer Fachmedien. ISBN 978-3-658-06993-3 <https://doi.org/10.1007/978-3-658-06993-3>
- IWANOWSKI, Sebastian und Rainer LANG, 2014. *Diskrete Mathematik mit Grundlagen: Lehrbuch für Studierende von MINT-Fächern*. Wiesbaden: Springer Fachmedien. ISBN 978-3-658-07131-8
- SCHULZE-PILLOT, Rainer, 2015. *Einführung in Algebra und Zahlentheorie*. 3. Auflage. Berlin: Springer Spektrum. ISBN 978-3-642-55215-1

**Anmerkungen:**

Keine Anmerkungen

<b>Computergrafik</b>			
<b>Modulkürzel:</b>	IB_CG	<b>SPO-Nr.:</b>	21
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	4
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Sommersemester
<b>Modulverantwortliche(r):</b>	Passig, Georg		
<b>Leistungspunkte / SWS:</b>	5 ECTS / 4 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	47 h	
	Selbststudium:	78 h	
	Gesamtaufwand:	125 h	
<b>Lehrveranstaltungen des Moduls:</b>	21.1: Computergrafik 21.2: Praktikum Computergrafik		
<b>Lehrformen des Moduls:</b>	21.1: SU/Ü - seminaristischer Unterricht/Übung 21.2: Pr - Praktikum		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
21.1: schrP90 - schriftliche Prüfung, 90 Minuten 21.2: LN - ohne/mit Erfolg teilgenommen  Weitere Erläuterungen: Erfolgreiche Abgabe von 5 praktischen Aufgaben über das automatische Testsystem.			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Sicherer Umgang mit der Programmiersprache Java. Sicherer Umgang mit Matrizen, Determinanten und linearen Abbildungen			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch der Vorlesung sind die Studierenden in der Lage, <ul style="list-style-type: none"> <li>• die wichtigsten Konzepte, Algorithmen und Methoden der Computergrafik zu verstehen und grundlegende Techniken der Computergrafik anzuwenden</li> <li>• 2D-Grafik und Rasterisierung anzuwenden</li> <li>• Graustufen und Farbdarstellung zu erklären</li> <li>• den Zweck von Bézier-Kurven, B-Splines, NURBS zu verstehen und in Anwendungen einzusetzen</li> <li>• Bestandteile der 3D-Modellierung zu benennen und Techniken der 3D-Modellierung zu verwenden</li> <li>• problemabhängig Techniken der Oberflächendarstellung anzuwenden</li> <li>• Transformationen und Projektionen einzusetzen</li> <li>• 3D-Welt- zu 2D-Gerätekoordinaten zu verwenden</li> <li>• Verschiedene Algorithmen für Sichtbarkeitsentscheide zu benennen und zu verwenden</li> </ul>			

- Techniken für Beleuchtungs-, Schattierungs- und Schatteneffekte einzusetzen
  - Unterschiedliche Abbildungen (Mappings) zur Darstellung von Texturen zu verwenden
- Praktikum:
- einfache computersimulierte 3D-Welten zu generieren und mit selbst erstellter Software zu rendern
  - den Zusammenhang der Komponenten der Rendering-Pipeline zu verstehen
  - anwendungsorientierte Programmieraufgaben auf dem Gebiet der Computergrafik in Java auszuführen

**Inhalt:**

- 2D-Grafik und Rasterisierung: Koordinatensysteme, Liniendarstellung, Aliasing, Polygonrasterung, Clipping, Transformationen
- Graustufen und Farbdarstellung
- Kurven: Bézier-Kurven, B-Splines, NURBS
- 3D-Modellierung: Szenegraph, Volumenmodellierung, Normzellen, Octrees, CSG, Kanten- und Oberflächenmodellierung, Drahtmodell
- Oberflächendarstellung: Topologische Struktur, Euler-Operationen, Flächen, Polygone, bilineare Interpolation, Coon's Patch, Freiformflächen
- Transformationen und Projektionen: Affine Transformationen, perspektivische und projektive Abbildungen
- 3D-Welt- zu 2D-Gerätekoordinaten: Ansichtstransformation, Projektionstransformation, Clipping, Normalisierung, Bildschirmtransformation
- Sichtbarkeitsentscheid: 3D-Clipping, Rückseitenentfernung, Tiefenpuffer-Algorithmus, List-Priority-Verfahren, Scan-Line-Verfahren, Raycasting
- Beleuchtung und Schattierung: Ambiente, diffuse, spekulare Reflexion, Entfernungsabhängigkeit, Nebel, Schattierung, Transparenz
- Schattenberechnung: Schattenvolumen, Schatten-Tiefenpuffer
- Globale Beleuchtungsmodelle: Raytracing, Radiosity
- Texturen: Texture Mapping, Texturkoordinatensystem, Two-Pass Mapping, Environment Mapping, Bump Mapping, Aliasing

**Literatur:**

- BENDER, Michael und Manfred BRILL, 2006. *Computergrafik: ein anwendungsorientiertes Lehrbuch*. 2. Auflage. München [u.a.]: Hanser. ISBN 3-446-40434-1
- HUGHES, John F., Andries VAN DAM und Morgan MCGUIRE, 2013. *Computer Graphics*. 3. Auflage. ISBN 978-0321-39952-6
- AKENINE-MÖLLER, Tomas, Eric HAINES und Naty HOFFMAN, 2018. *Real-time Rendering*. F. Auflage. ISBN 978-1-138-62700-0
- LENGYEL, Eric, 2019. *Foundations of Game Engine Development, Volume 2: Rendering*. ISBN 978-0985811754

**Anmerkungen:**

Keine Anmerkungen

<b>Projektmanagement</b>			
<b>Modulkürzel:</b>	IB_PM	<b>SPO-Nr.:</b>	22
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	4
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Sommersemester
<b>Modulverantwortliche(r):</b>	Regensburger, Franz		
<b>Leistungspunkte / SWS:</b>	5 ECTS / 4 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	47 h	
	Selbststudium:	78 h	
	Gesamtaufwand:	125 h	
<b>Lehrveranstaltungen des Moduls:</b>	Projektmanagement		
<b>Lehrformen des Moduls:</b>	SU/Ü - seminaristischer Unterricht/Übung		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
schrP90 - schriftliche Prüfung, 90 Minuten			
Weitere Erläuterungen: Keine			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Vorerfahrung bei der Planung und Durchführung einfacher Projekte			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch des Moduls sind die Studierenden in der Lage,			
<ul style="list-style-type: none"> <li>• das Management für kleinere und mittlere Projekte im industriellen/technischen Umfeld zu übernehmen</li> <li>• die relevanten Schritte in der Vorphase der Planungen eines Projekts zu benennen und selbständig durchzuführen</li> <li>• Ergebnisse in einer kurzen Präsentation vorzustellen und zu diskutieren</li> <li>• einen korrekten Start (Kick-off) eines Projekts zu organisieren und alle dafür erforderlichen Vorarbeiten und Analysen zu erledigen</li> <li>• ein Projekt im Detail zu planen und haben dies auch an einem realen Fall durchgeführt</li> <li>• mehrere Methoden zur Analyse eines laufenden Projekts und zur Erstellung von Trendaussagen über den Fortschritt des Projekts zu benennen</li> <li>• relevante Zusammenhänge im Ablauf von Projekten zu verstehen und, basierend auf fundierten Methoden, Entscheidungen für die weitere Steuerung eines Projekts zu fällen</li> <li>• neue Ansätze und Methoden des agilen Projektmanagements einzusetzen</li> </ul>			

<ul style="list-style-type: none"> <li>• Basis-Techniken des wissenschaftlichen Arbeitens einzusetzen</li> </ul>
<b>Inhalt:</b>
<ul style="list-style-type: none"> <li>• Grundlagen: Definition Projekt, Projektdreiecks (Zeit, Budget, Leistung); Typische Projektorganisationen</li> <li>• Vorphase eines Projekts: Vorgehensmodelle; Zieldefinition; Stakeholder-Analyse / -Management; Risiko-Analyse / -Management; Scope und Kick-off; Gruppenarbeiten zur Vertiefung</li> <li>• Planung eines Projekts: Projektstrukturplan, Ablaufplan / Netzpläne; Aufwandschätzungen; Ressourcenplanung</li> <li>• Durchführung eines Projekts: Fortschritt- und Trend-Analysen; Kosten / Berichterstattung; Controlling und Änderungsmanagement</li> <li>• Agile Methoden des Projektmanagements: Idee und Ansatz agiler Methoden im Projektmanagement; Vorgehen und Rollen bei Scrum</li> <li>• Zusätzlichen Modul zum wissenschaftlichen Arbeiten: Recherche und Quellen: Recherchestrategie, Evaluation der Informationsquellen; Richtiges Zitieren für wissenschaftliche Arbeiten, Plagiate</li> </ul>
<b>Literatur:</b>
<ul style="list-style-type: none"> <li>• SEIBERT, Siegfried, 1998. <i>Technisches Management : Innovationsmanagement, Projektmanagement, Qualitätsmanagement</i>. 1. Auflage. Stuttgart : Teubner. ISBN 3-519-06363-8</li> <li>• BOHINC, Tomas, 2019. <i>Grundlagen des Projektmanagements : Methoden, Techniken und Tools für Projektleiter</i>. 7. Auflage. Offenbach am Main: GABAL. ISBN 978-3-86936-912-9</li> <li>• SUTHERLAND, Jeffrey Victor , 2015. <i>Die Scrum-Revolution: Management mit der bahnbrechenden Methode der erfolgreichsten Unternehmen</i>. 1. Auflage. Frankfurt am Main: Campus. ISBN 978-3-593-39992-8 ; 3-593-39992-X</li> <li>• SCHELLE, Heinz und Roland OTTMANN, 2014. <i>Projekte zum Erfolg führen: projektmanagement systematisch und kompakt</i>. 7. Auflage. München: C.H.Beck. ISBN 978-1-4619-5865-9, 1-4619-5865-2</li> </ul>
<b>Anmerkungen:</b>
Keine Anmerkungen



<b>Vorbereitendes Praxisseminar (PLV1)</b>			
<b>Modulkürzel:</b>	IB_PLV1	<b>SPO-Nr.:</b>	23
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	5
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	Winter- und Sommersemester
<b>Modulverantwortliche(r):</b>	Regensburger, Franz		
<b>Leistungspunkte / SWS:</b>	2 ECTS / 1 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	12 h	
	Selbststudium:	38 h	
	Gesamtaufwand:	50 h	
<b>Lehrveranstaltungen des Moduls:</b>	Vorbereitendes Praxisseminar (PLV1)		
<b>Lehrformen des Moduls:</b>	S - Seminar		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
LN - ohne/mit Erfolg teilgenommen			
Weitere Erläuterungen: Anwesenheit und aktive Mitarbeit bei Einzel- und Gruppenübungen sowie Rollenspielen (unbenotet).			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Es existiert bereits eine Vorstellung darüber, in welchem Betrieb das Praktikum absolviert werden soll.			
<b>Angestrebte Lernergebnisse:</b>			
Nach erfolgreicher Teilnahme an den Modulveranstaltungen sind die Studierenden in der Lage, <ul style="list-style-type: none"> <li>• sich in alltäglichen Situationen des beruflichen Miteinanders angemessen zu verhalten</li> <li>• ihre eigene Kommunikations- und Teamkompetenz zu reflektieren und gezielter einzusetzen</li> <li>• Konflikte und deren Dynamik zu analysieren</li> <li>• zielführende Lösungsansätze im Umgang mit kritischen Situationen und Konflikten zu entwickeln</li> <li>• Kritik an der eigenen Person anzunehmen und zu reflektieren</li> <li>• Faires Feedback zum Verhalten anderer Personen zu geben</li> </ul>			
<b>Inhalt:</b>			
<ul style="list-style-type: none"> <li>• Diskussion von Erwartungen, Befürchtungen, Unsicherheiten und Handlungsempfehlungen im Hinblick auf das bevorstehende Firmenpraktikum</li> <li>• Einschätzung von Persönlichkeitsprofilen</li> <li>• Reflexion eigener Stärken und Schwächen</li> </ul>			

- Einüben verschiedener Kommunikations- und Konfliktlösungstechniken im Rahmen von Gruppenübungen und Rollenspielen
- Anderen Personen konstruktives Feedback geben
- Kritik an der eigenen Person annehmen

**Literatur:**

- GAY, Friedbert, 2004. *Das persolog Persönlichkeits-Profil: Persönliche Stärke ist kein Zufall.*

**Anmerkungen:**

Für Dual-Studierende mit beruflicher Praxis kann die Veranstaltung angerechnet werden.

<b>Praktikum</b>			
<b>Modulkürzel:</b>	IB_Prak	<b>SPO-Nr.:</b>	24
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	5
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Wintersemester
<b>Modulverantwortliche(r):</b>	Hafenrichter, Bernd		
<b>Leistungspunkte / SWS:</b>	24 ECTS / 0 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	0 h	
	Selbststudium:	600 h	
	Gesamtaufwand:	600 h	
<b>Lehrveranstaltungen des Moduls:</b>	Praktikum		
<b>Lehrformen des Moduls:</b>	Pr - Praktikum		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden		
<b>Prüfungsleistungen:</b>			
PB - Praktikumsbericht			
Weitere Erläuterungen: Abgabe eines Praktikumsberichts (PrB). Siehe Anlage SPO			
<b>Voraussetzungen gemäß SPO:</b>			
Das Praktikum des praktischen Studiensemesters darf nur belegt werden, wenn alle Prüfungen des ersten Studienabschnitts bestanden und mindestens 20 Leistungspunkte aus Modulen der ersten beiden Semester des zweiten Studienabschnitts erzielt wurden.			
<b>Empfohlene Voraussetzungen:</b>			
Das Modul Vorbereitendes Praxisseminar (PLV1) wurde bereits besucht			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch des Moduls			
<ul style="list-style-type: none"> <li>• können die Studierende im Studium wissenschaftliche Methoden auf konkrete Problemstellungen der Informatik anwenden</li> <li>• sind den Studierenden die zukünftigen beruflichen Anforderungen bekannt</li> <li>• kennen die Studierenden die grundlegenden Elemente des betrieblichen Alltags</li> <li>• können die Studierenden eigenverantwortlich Aufgaben bzw. Teilaufgaben, die auf den Studienfortschritt abgestimmt sind, erfolgreich lösen</li> <li>• können die Studierenden als Mitglied eines Projektteams zum Gesamterfolg beitragen</li> </ul>			
Im Rahmen des Praktikums sollen praktische Tätigkeiten im Berufsbild eines Informatikers vertieft werden. Unter anderem gehört hierzu folgendes:			
<ul style="list-style-type: none"> <li>• IT-Systeme entwerfen</li> <li>• IT-Systeme implementieren</li> <li>• IT-Systeme beschaffen, integrieren und betreuen</li> </ul>			

<ul style="list-style-type: none"><li>• IT-Systeme testen</li></ul>
<b>Inhalt:</b>
<ul style="list-style-type: none"><li>• Auswahl eines geeigneten Unternehmens im In- oder Ausland</li><li>• Mitarbeit an konkreten betrieblichen Aufgabenstellungen unter Anwendung der erlernten wissenschaftlichen Methoden</li><li>• Aktive Teilnahme an Team-Sitzungen mit Beiträgen zum Arbeitsfortschritt</li><li>• Exploration der Entwicklungswerkzeuge, die im Unternehmen verwendet werden (IDE, RCS, Ticketsystem, CI/CD)</li><li>• Erstellen eines Arbeitsplanes für das Praktikum mit definierten eigenverantwortlich zu bearbeitenden Arbeitspaketen</li><li>• Erstellen eines Praktikumsberichtes</li></ul>
<b>Literatur:</b>
<ul style="list-style-type: none"><li>• Ohne Autor. <i>Allgemeine Informationen Fakultät Informatik - Praxissemester</i> [online]. [Zugriff am: ]. Verfügbar unter: <a href="https://moodle.thi.de/course/view.php?id=6204&amp;section=3">https://moodle.thi.de/course/view.php?id=6204&amp;section=3</a></li><li>• Ohne Autor. <i>Wichtige Informationen des SCS zum Praxissemester</i> [online]. [Zugriff am: ]. Verfügbar unter: <a href="https://moodle.thi.de/course/view.php?id=6758">https://moodle.thi.de/course/view.php?id=6758</a></li></ul>
<b>Anmerkungen:</b>
Dual-Studierende müssen gemäß APO §29 (3) das Praxissemester bei ihrem Dual-Unternehmen ableisten.

<b>Nachbereitendes Praxisseminar (PLV 2)</b>			
<b>Modulkürzel:</b>	IB_PLV2	<b>SPO-Nr.:</b>	25
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	5
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	Winter- und Sommersemester
<b>Modulverantwortliche(r):</b>	Fuchs, Hildegard		
<b>Leistungspunkte / SWS:</b>	2 ECTS / 1 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	12 h	
	Selbststudium:	38 h	
	Gesamtaufwand:	50 h	
<b>Lehrveranstaltungen des Moduls:</b>	Nachbereitendes Praxisseminar (PLV 2)		
<b>Lehrformen des Moduls:</b>	S - Seminar		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
LN - ohne/mit Erfolg teilgenommen			
Weitere Erläuterungen: Um an der Veranstaltung mit Erfolg teilzunehmen, muss jeder Teilnehmer / jede Teilnehmerin ein Kurzreferat (mind. 15 bis max. 20 Minuten) halten.			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Das Betriebspraktikum wurde bereits absolviert.			
<b>Angestrebte Lernergebnisse:</b>			
Nach erfolgreicher Teilnahme an der Lehrveranstaltung sind die Studierenden in der Lage, <ul style="list-style-type: none"> <li>• ihre eigenen Projekterfahrungen in Relation zu denen anderer Studierenden zu reflektieren</li> <li>• ihre Präsentationsleistung durch das Feedback der anderen Teilnehmer objektiv einzuschätzen</li> </ul> Die erfolgreiche Teilnahme ermöglicht ihnen, <ul style="list-style-type: none"> <li>• ihre Erkenntnisse durch moderierte Diskussion, Anleitung und Beratung zu vertiefen und zu sichern</li> <li>• ihre Erfahrungen aus der Praxis mit theoretischen Kenntnissen zu verbinden</li> <li>• vertiefte Kenntnisse über die Vielfalt möglicher Lösungsansätze zu typischen fachlichen und methodischen Problemstellungen zu erwerben</li> <li>• ihre Sozialkompetenz zu stärken</li> </ul>			
<b>Inhalt:</b>			
<ul style="list-style-type: none"> <li>• Einführung</li> </ul>			

- Präsentation der Themen in Kurzreferaten (jeweils mind. 15 bis max. 20 Minuten)
- Anschließende Diskussion der Inhalte und Aussagen des Referats
- Anschließende Diskussion der Darbietung des Referenten
- Konstruktives Feedback geben
- Kritik an der eigenen Person zulassen und annehmen

**Literatur:**

- BEUSHAUSEN, Ulla, 2023. *Sicher und frei reden: Sprechängste erfolgreich abbauen - Trainingsprogramm mit 6 Bausteinen*. 5. Auflage. München: Ernst Reinhardt Verlag. ISBN 978-3-497-03206-8

**Anmerkungen:**

Für Dual-Studierende ist eine spezielle Veranstaltung PLV 2 eingeplant.

<b>Informations- und Medienkompetenz (PLV3)</b>			
<b>Modulkürzel:</b>	IB_PLV3	<b>SPO-Nr.:</b>	26
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	5
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	Winter- und Sommersemester
<b>Modulverantwortliche(r):</b>	Regensburger, Franz		
<b>Leistungspunkte / SWS:</b>	2 ECTS / 1 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	6 h	
	Selbststudium:	44 h	
	Gesamtaufwand:	50 h	
<b>Lehrveranstaltungen des Moduls:</b>	Informations- und Medienkompetenz (PLV3)		
<b>Lehrformen des Moduls:</b>	S - Seminar		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
LN - ohne/mit Erfolg teilgenommen			
Weitere Erläuterungen: Um an der Veranstaltung mit Erfolg teilzunehmen, sind die E-Learning-Anteile verpflichtend, ebenso die Anwesenheit in den Live-Meetings.			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Es besteht bereits eine Vorstellung, welches Thema die eigene Bachelorarbeit haben soll.			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Absolvieren des Kurses sind die Studierenden in der Lage, <ul style="list-style-type: none"> <li>• die unterschiedlichen Möglichkeiten des Publizierens zu beschreiben</li> <li>• geeignete Methoden der Informationsrecherche auszuwählen</li> <li>• systematisch und zielorientiert eine Informationsrecherche durchzuführen</li> <li>• mit unterschiedlichen Informationssystemen für die Literaturrecherche umzugehen</li> <li>• einschlägige Vorschriften des Zitierens anzuwenden und ein dazu passendes Literaturverzeichnis zu erstellen</li> <li>• Werkzeuge anzuwenden, mit deren Hilfe sie Literaturquellen verwalten und Literaturverzeichnisse erstellen können</li> </ul>			

**Inhalt:**

Anhand eines vorgegebenen Themas, vorzugsweise das Thema der eigenen Bachelorarbeit, erarbeiten sich die Studierenden in kleinen Teams Strategien der Informationsrecherche und trainieren die wichtigsten Rechercheinstrumente für ihr Fachgebiet.

- Wege des wissenschaftlichen Publizierens
- Methodik der Informationsrecherche
- Ablauf der systematischen und zielorientierten Recherche
- Die wichtigsten Instrumente für das Fachgebiet: Bibliothekskataloge, Fernleihe, Wissenschaftliche Fachdatenbanken, Normen und Patente
- Wissenschaftliches Arbeiten: Zitieren, Literaturverzeichnis
- Werkzeuge für die Literaturverwaltung

**Literatur:**

- BALZERT, Helmut, Marion SCHRÖDER und Christian SCHÄFER, 2017. *Wissenschaftliches Arbeiten: Ethik, Inhalt & Form wiss. Arbeiten, Handwerkszeug, Quellen, Projektmanagement, Präsentation*. 2. Auflage. Berlin ; Dortmund: Springer Campus. ISBN 978-3-96149-006-6
- SANDBERG, Berit, 2017. *Wissenschaftliches Arbeiten von Abbildung bis Zitat: Lehr- und Übungsbuch für Bachelor, Master und Promotion*. 3. Auflage. Berlin: De Gruyter Oldenbourg. ISBN 978-3-11-051477-3, 3-11-051477-X

**Anmerkungen:**

Diese Veranstaltung sollte kurz vor dem Beginn der Bachelorarbeit absolviert werden. Dadurch werden die Motivation für die Teilnahme und der Nutzen der Veranstaltung wesentlich gesteigert. Die Veranstaltung findet ausschließlich online während der Vorlesungszeit statt.



<b>Programmierparadigmen und deren Sprachen</b>			
<b>Modulkürzel:</b>	IB-PPS	<b>SPO-Nr.:</b>	27
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	7
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Wintersemester
<b>Modulverantwortliche(r):</b>	Regensburger, Franz		
<b>Leistungspunkte / SWS:</b>	5 ECTS / 4 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	47 h	
	Selbststudium:	78 h	
	Gesamtaufwand:	125 h	
<b>Lehrveranstaltungen des Moduls:</b>	Programmierparadigmen und deren Sprachen		
<b>Lehrformen des Moduls:</b>	SU/Ü - seminaristischer Unterricht/Übung		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
schrP90 - schriftliche Prüfung, 90 Minuten			
Weitere Erläuterungen: Keine			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Fundierte Kenntnisse und praktische Erfahrung mit mindestens einer prozeduralen, einer objektorientierten, einer logischen Programmiersprache (z.B. PROLOG). Vorteilhaft sind auch Kenntnisse in der funktionalen Programmierung. Sicherer Umgang mit logischen Formeln, sicherer Umgang mit Grammatiken für kontextfreie Sprachen			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch des Moduls sind die Studierenden in der Lage,			
<ul style="list-style-type: none"> <li>wesentliche Programmierparadigmen zu unterscheiden und deren Unterschiede sowie Vor- und Nachteile zu benennen</li> <li>die Vorzüge der higher-order Programmierung zu erkennen und auf geeignete Problemstellung anzuwenden</li> <li>den Unterschied zwischen reinen und nicht-reinen funktionalen Programmiersprachen zu erklären</li> <li>den Unterschied zwischen getypten und ungetypten Programmiersprachen zu erklären</li> <li>den vorteilhaften Einfluss von Typinformation auf die Code-Optimierung zu verstehen</li> <li>den Zusammenhang zwischen funktionalen Sprachen und dem ihnen zugrundeliegenden Lambda-Kalkül zu erklären</li> </ul>			

<ul style="list-style-type: none"><li>• den Unterschied zwischen lazy und eager Evaluationsstrategien zu beschreiben und problemabhängig die adäquate Strategie einzusetzen</li><li>• einfache Gleichungsketten im ungetypten Lambda-Kalkül zu bilden, um dadurch unterschiedliche Reduktionsstrategien zu demonstrieren</li><li>• Programme in der stark getypten funktionalen Sprachen Haskell zu erstellen</li><li>• Parser für kontextfreie Sprachen unter Einsatz monadischer higher-order Parser-Kombinatoren in der Sprache Haskell zu implementieren</li></ul>
<b>Inhalt:</b>
<ul style="list-style-type: none"><li>• Überblick über gängige Programmierparadigmen: prozedurale, objektorientierte, funktionale, logische Programmierung</li><li>• Grundlagen: der ungetypte Lambda-Kalkül</li><li>• Einführung in die funktionale Programmierung mit Haskell</li><li>• Evaluationsstrategien (eager und lazy)</li><li>• Ungetypte funktionale Sprachen: Scheme</li><li>• Getypte funktionale Sprachen: Haskell, Standard ML, Scala</li><li>• Realisierung von Konstrukten des Lambda-Kalküls in herkömmlichen Sprachen wie Java und Python</li><li>• Implementierung von Parsern durch monadische Parser-Kombinatoren</li></ul>
<b>Literatur:</b>
<ul style="list-style-type: none"><li>• PAULSON, Larry, 1996. <i>ML for the working programmer</i>. 2. Auflage. ISBN 978-0-521-56543-1</li><li>• HUTTON, Graham, 2007. <i>Programming in Haskell</i>. 2. Auflage. ISBN 978-1-316-62622-1</li><li>• SULLIVAN, GOERZEN und STEWART, 2008. <i>Real World Haskell</i>. ISBN 987-0-596-51498-3</li></ul>
<b>Anmerkungen:</b>
Keine Anmerkungen

<b>Statistik und Kombinatorik</b>			
<b>Modulkürzel:</b>	IB_SC	<b>SPO-Nr.:</b>	28
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	6
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Sommersemester
<b>Modulverantwortliche(r):</b>	Lorencka, Joanna		
<b>Leistungspunkte / SWS:</b>	5 ECTS / 4 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	47 h	
	Selbststudium:	78 h	
	Gesamtaufwand:	125 h	
<b>Lehrveranstaltungen des Moduls:</b>	Statistik und Kombinatorik		
<b>Lehrformen des Moduls:</b>	SU/Ü - seminaristischer Unterricht/Übung		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
schrP90 - schriftliche Prüfung, 90 Minuten			
Weitere Erläuterungen: Keine			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Grundlagen der Mathematik 1 und 2			
<b>Angestrebte Lernergebnisse:</b>			
<p>Nach erfolgreicher Teilnahme an den Modulveranstaltungen sind die Studierenden in der Lage,</p> <ul style="list-style-type: none"> <li>• zwischen Permutation, Kombination und Auswahl zu unterscheiden</li> <li>• das Einschluss-Ausschluss-Prinzip wiederzugeben und anzuwenden</li> <li>• das Schubfach-Prinzip zu verstehen und anzuwenden</li> <li>• erzeugende Funktionen zu definieren und einzusetzen</li> <li>• die Kombinatorik auf die Wahrscheinlichkeitsrechnung anzuwenden</li> <li>• die Verfahren der deskriptiven Statistik anzuwenden</li> <li>• die grundlegenden Konzepte der schließenden Statistik zu beschreiben</li> <li>• mit der Statistik-Software R grundlegend umzugehen</li> <li>• mit externen Datensätzen umzugehen, R-interne Datenobjekte anzulegen und R-Funktionen anzuwenden</li> <li>• statistische Methoden, insbesondere zur Datenvisualisierung, Stichprobenziehung sowie Schätz- und Testverfahren in R umzusetzen</li> </ul>			

**Inhalt:**

- Kombinatorische Anzahlprobleme: Permutationen, Kombination, Auswahl
- Ausschluss-Einschluss-Prinzip; Schubfach-Prinzip
- Erzeugende Funktionen in der Kombinatorik
- Wahrscheinlichkeitsrechnung: Zufallsexperimente, Ausgänge, Grundmenge, Ereignisse
- Deskriptive Statistik: Organisation und Darstellung von Daten, Skalentypen, Statistische Kennwerte, Standardisierung von Daten
- Inferenzstatistik: Die Normalverteilung, die Standardnormalverteilung, die Stichprobenkennwerteverteilung, Punkt- und Intervallschätzen, Konfidenzintervalle, Hypothesentesten, t-Test,  $\beta$ -Fehler und Teststärke, Konfidenzniveau, Effektgrößen
- Merkmalszusammenhänge: Kovarianz und Korrelation, einfache lineare Regression, die Güte des Zusammenhangs
- Das Grundprinzip der Varianzanalyse, Verfahren für Rang- und Nominaldaten

**Literatur:**

- RASCH, Björn, 2014. *Quantitative Methoden: Band 1 u. 2*. 4. Auflage. Berlin: Springer.
- SEDLMEIER, Peter und Frank RENKEWITZ, 2018. *Forschungsmethoden und Statistik für Psychologen und Sozialwissenschaftler*. 3. Auflage. ISBN 978-3-86894-321-4
- HATZINGER, Reinhold, Kurt HORNIK und Herbert NAGEL, 2014. *R: Einführung durch angewandte Statistik*. 2. Auflage.
- WEWEL, Max C., 2019. *Statistik im Bachelor-Studium der BWL und VWL: Methoden, Anwendungen, Interpretation*. 4. Auflage. Hallbergmoos/Germany: Pearson. ISBN 978-3-86894-381-8
- TITTMANN, Peter, 2019. *Einführung in die Kombinatorik* [online]. Berlin ; Heidelberg: Springer Spektrum PDF e-Book. ISBN 978-3-662-58921-2. Verfügbar unter: <https://doi.org/10.1007/978-3-662-58921-2>.

**Anmerkungen:**

Keine Anmerkungen

<b>Projekt</b>			
<b>Modulkürzel:</b>	IB_PRJ	<b>SPO-Nr.:</b>	29
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	6
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	Winter- und Sommersemester
<b>Modulverantwortliche(r):</b>	Regensburger, Franz		
<b>Leistungspunkte / SWS:</b>	5 ECTS / 4 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	47 h	
	Selbststudium:	78 h	
	Gesamtaufwand:	125 h	
<b>Lehrveranstaltungen des Moduls:</b>	Projekt		
<b>Lehrformen des Moduls:</b>	Prj - Projekt		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
PA - praktische Arbeit/Studienarbeit			
Weitere Erläuterungen:			
<ul style="list-style-type: none"> <li>• Bei der Projektarbeit handelt es sich um eine Gruppenarbeit, bei der mehrere Studierende eine gemeinsame Aufgabenstellung im Team bearbeiten.</li> <li>• Jeder Studierende hat zur gemeinsamen Aufgabenstellung individuell beizutragen, einen Projektbericht abzuliefern und ggf. die Ergebnisse mündlich zu präsentieren.</li> <li>• Umfang des Projektberichtes lt. APO: 1500 Wörter bis 7500 Wörter bzw. ca. 5 bis 25 Seiten</li> <li>• Umfang der mündlichen Präsentation lt. APO: 15 bis 45 Minuten.</li> <li>• Der Projektbericht ist mit einem Textverarbeitungsprogramm zu erstellen.</li> </ul>			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Beherrschung der gängigen Werkzeuge für die Software-Entwicklung: IDE mit Editor, Compiler, Debugger. Solide Kenntnisse in Programmiersprachen wie Java, Python, PHP, JavaScript, Go			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch des Moduls sind die Studierenden in der Lage,			
<ul style="list-style-type: none"> <li>• mindestens eine bestimmte Projektmanagementmethode praktisch einzusetzen (vorzugsweise Scrum oder Kanban)</li> <li>• konkrete Werkzeuge einzusetzen, die üblicherweise im Rahmen der Durchführung eines IT-Projekts zur Anwendung kommen (IDE, RCS, Ticket-System, Scrum Board)</li> </ul>			

- mit fachlichen und nicht-fachlichen Problemen umzugehen, die während der Durchführung eines mehrwöchigen Projekts auftreten können
- eine komplexe fachliche Aufgabenstellung zu analysieren und über ein Semester hinweg in einem Team erfolgreich zu bearbeiten
- in unterschiedlicher aber stets angemessener Ausführlichkeit über den Projektfortschritt in mündlicher und/oder schriftlicher Form zu berichten
- neben fachlichen Problemstellungen auch betriebswirtschaftliche Aspekte eines Projekts zu erkennen und diese zur Gewährleistung des Gesamterfolgs angemessen zu unterstützen

**Inhalt:**

- Praktische Anwendung einer Projektmanagement-Methode (z.B. Scrum, Kanban)
- Planen von Aufgaben (Tasks) und Aufwandsabschätzung im Team
- Praktische Anwendung von Software-Entwicklungswerkzeugen (IDE, RCS, Ticket System, Scrum Board)
- Arbeiten im Team
- Einsatz von Techniken der agilen Software-Entwicklung: Pair Programming, Test Driven Development, Unit Testing, CI/CD
- Reporting des Projektfortschritts, z.B. Daily Scrum
- Präsentation von Projektergebnissen

**Literatur:**

- SHORE, James und Shane WARDEN, 2007. *The Art of Agile Development*. ISBN 978-0596527679
- SCHWABER, Ken und Mike BEEDLE, 2002. *Agile Software Development with Scrum*. ISBN 0-13-207489-3
- WILSON, Redmond, 2012. *Seven Databases in seven Weeks*. ISBN 978-1-93435-692-0

**Anmerkungen:**

Im Allgemeinen werden die Projekte in Kooperation mit externen Firmen oder dem hochschuleigenen Forschungszentrum durchgeführt. Alternativ können auch Dozenten gezielt Projektthemen vorgeben, die im Rahmen ihrer Lehr- oder Forschungstätigkeit bearbeitet werden sollen.

Die Projektleitung und die Organisation werden von Studierenden ausgeführt. Der Dozent/Lehrbeauftragte fungiert lediglich als Coach und/oder Auftraggeber.

Als Projektmanagementmethode können klassische Methoden oder agile Methoden wie Scrum oder Kanban verwendet werden. Die Entscheidung darüber, welche Methode verwendet wird, liegt beim Projektteam.

Zu Beginn des Projekts kommuniziert der Dozent/Lehrbeauftragte klar seine Erwartungen hinsichtlich Termine, Form und Nachweis der individuellen Leistungen, die von allen Studierenden zu erbringen sind.

Das Projektteam einigt sich mit dem Dozenten/Lehrbeauftragten über die Kommunikations- und Dokumentationsformen, die während der Projektlaufzeit von allen Projektteilnehmern (Studierende, Dozent, Auftraggeber) einzuhalten sind.

Zu klären sind:

- Häufigkeit und Dauer von Planungssitzungen
- Art und Durchführung der Treffen (gemeinsam oder virtuell/elektronisch)
- Turnusmäßige Treffen (evtl. täglich in Form von Scrum-Meetings etc.)
- Art und Umfang der Deliverables
- Art und Umfang der individuellen Beiträge durch Studierende
- Kriterien für die Beurteilung/Benotung durch den Dozenten

Das Projekt ist besonders für Dual-Studierende geeignet.

<b>IT-Recht</b>			
<b>Modulkürzel:</b>	IB_ITR	<b>SPO-Nr.:</b>	30
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	6
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	Winter- und Sommersemester
<b>Modulverantwortliche(r):</b>	Regler, Michaela		
<b>Leistungspunkte / SWS:</b>	5 ECTS / 4 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	47 h	
	Selbststudium:	78 h	
	Gesamtaufwand:	125 h	
<b>Lehrveranstaltungen des Moduls:</b>	IT-Recht		
<b>Lehrformen des Moduls:</b>	SU/Ü - seminaristischer Unterricht/Übung		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
schrP90 - schriftliche Prüfung, 90 Minuten			
Weitere Erläuterungen: Keine			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Grundlegende Kenntnis unterschiedlicher Software-Lizenzmodelle			
<b>Angestrebte Lernergebnisse:</b>			
Nach erfolgreicher Teilnahme an der Lehrveranstaltung sind die Studierenden in der Lage, <ul style="list-style-type: none"> <li>• wichtige Bereiche des Informationstechnologierechts zu beschreiben</li> <li>• wichtige Bereiche des Informationstechnologierechts in digitalen Geschäftsmodellen zu beachten</li> <li>• in ihrem Berufsfeld rechtlich relevante Probleme zu erkennen</li> <li>• Lösungsansätze zu erarbeiten und diese in der Praxis umzusetzen und anzuwenden.</li> <li>• mehrere Software-Lizenzmodelle zu verstehen und deren Unterschiede zu erklären</li> <li>• wesentliche Aspekte des Datenschutzes zu benennen und zu differenzieren</li> </ul>			
<b>Inhalt:</b>			
<ul style="list-style-type: none"> <li>• Überblick und Einordnung, Definition IT-Recht national und international</li> <li>• Vertragsschluss und Rechtsgeschäftslehre, Vertragspartner, Vertretung, Anfechtung, Widerruf, Methodik der Rechtsfindung</li> </ul>			

- Vertragsarten im Internet (Kauf-, Werk-, Dienst- und Mietverträge sowie Verträge eigener Art), Pflichten im elektronischen Rechtsverkehr
- Grundzüge Fernabsatzgeschäft (wichtige Pflichtangaben bei Shops, Widerrufsrecht, digitale Inhalte, Grundzüge AGB, OS-Plattform)
- Pflichtangaben nach TMG und DL-Info bei Shops, Apps, Social Media (Impressum, Haftung für Links, "Disclaimer" bei Shops und Verkaufsspalformen, UWG, PAngV)
- Einblick in das Domainrecht (Vergabe, Marken- und Namensrecht, Unterlassungs-, Schadensersatz- und Auskunftsanspruch, Gerichtszuständigkeit)
- Grundzüge des Urheberrechts und KunstUrR-Recht am eigenen Bild
- Umgang mit Bewertungsplattformen (Reaktion auf Bewertungen, Abgrenzung Meinungsäußerung/ Tatsachenbehauptung, Löschung von Bewertungen)
- Computerkriminalität / Strafrecht
- Verfahrensrecht (Mahnverfahren, Abmahnung, Klage und einstweilige Verfügung) und Umgang mit Behörden und Beratern

**Literatur:**

- REDEKER, Helmut, 2020. *IT-Recht*. 7. Auflage. München: C.H. Beck. ISBN 978-3-406-68727-3, 3-406-68727-X
- SCHNEIDER, Jochen, 2020. *IT- und Computerrecht: Textausgabe*. 413. Auflage. München: dtv. ISBN 978-3-423-05562-8
- BENNING, Axel und Jörg-Dieter OBERRATH, 2008. *Computer- und Internetrecht*. 2. Auflage. Stuttgart ; München [u.a.]: Boorberg. ISBN 978-3-415-04067-0
- EHMANN, Eugen und Thomas BRUGGMANN, 2017. *Lexikon für das IT-Recht 2017/2018: Spezialausgabe für Behörden*. 5. Auflage. Heidelberg: Jehle. ISBN 978-3-7825-0606-9

**Anmerkungen:**

3 SWS für Präsenz und 1 SWS für digitale Lehrvideos zur Gerichtsbarkeit, Geschäftsfähigkeit, Vertragsfreiheit, Anfechtung, zum Widerrufsrecht im e-commerce, diversen Vertragsarten im BGB und zum Strafrecht im IT-Bereich



<b>IT-Sicherheit</b>			
<b>Modulkürzel:</b>	IB_ITS	<b>SPO-Nr.:</b>	32
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	6
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Sommersemester
<b>Modulverantwortliche(r):</b>	Hof, Hans-Joachim		
<b>Leistungspunkte / SWS:</b>	5 ECTS / 4 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	47 h	
	Selbststudium:	78 h	
	Gesamtaufwand:	125 h	
<b>Lehrveranstaltungen des Moduls:</b>	IT-Sicherheit		
<b>Lehrformen des Moduls:</b>	SU/Ü - seminaristischer Unterricht/Übung		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
schrP90 - schriftliche Prüfung, 90 Minuten			
Weitere Erläuterungen: Keine			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Erfahrung mit der Installation von Software-Paketen auf dem eigenen Rechner. Erfahrung in der Bedienung eines Internet-Browsers und der Einstellung sicherheitsrelevanter Parameter (Akzeptieren von Cookies, Lizenzen, Speicherung von Passwörtern im Browser).			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch des Moduls			
<ul style="list-style-type: none"> <li>• kennen die Studierenden die wesentlichen Sicherheitsrisiken und können deren Gefahrenpotential einschätzen</li> <li>• haben die Studierenden die Fähigkeit, selbst eine Risikoanalyse und deren Bewertung durchzuführen</li> <li>• sind die Studierenden in der Lage, anhand einer solchen Risikoanalyse eine geeignete IT-Sicherheitsstrategie zu entwerfen und dabei sowohl organisatorische als auch technische Aspekte zu berücksichtigen</li> <li>• sind die Studierenden in der Lage, die Wirksamkeit von Sicherheitsstrategien für die Praxis zu beurteilen</li> <li>• kennen die Studierenden die wichtigsten Verfahren zur Datenverschlüsselung und zur Authentifizierung</li> <li>• verstehen sie Studierenden die wesentlichen mathematischen Konzepte, die den Verfahren zur Datenverschlüsselung und zur Authentifizierung zugrunde liegen</li> </ul>			

<b>Inhalt:</b>
<ul style="list-style-type: none"><li>• Organisation und Sicherheitsmanagement: Sicherheitsmodelle und Sicherheitspolicies; Risikoanalyse von IT-Struktur und IT-gestützten Geschäftsprozessen</li><li>• Netzwerksicherheit: Technischer Hintergrund zu Netzwerkrisiken; Sicherheitstests von Netzen (mit Tools, z.B. Nessus, manuelle Tests); Schadprogramme und deren Abwehr; Vorgehensweise von Hacker und Crackern; Konzeption von Firewalls und Netzen; Intrusion Detection und Monitoring</li><li>• Betriebssysteme sicherer machen; Sichere Konfiguration; „Härtung“ eines Betriebssystems (Verwendung von Capabilities, durch Tools, z.B. Prozessabschottung mit AppAmor); Rootkits</li><li>• Zugriffsschutzsysteme: Authentifizierung, verschiedene Verfahren; Rechtesysteme auf unterschiedlichen Ebenen</li><li>• Softwarebedingte Schwachstellen und deren Vermeidung: Sichere Softwareentwicklung; Typische Schwachstellen wie Buffer- und Heapoverflows</li><li>• Mathematischer Background: Modulare Arithmetik; Endliche Gruppen; Endliche Körper</li><li>• Symmetrische Verschlüsselungsverfahren (z.B. AES)</li><li>• Public Key – Verfahren: RSA-Verfahren; Diffie-Hellman-Schlüsselaustausch; Signaturen</li></ul>
<b>Literatur:</b>
<ul style="list-style-type: none"><li>• MCNAB, Chris, 2016. <i>Network Security Assessment</i>.</li><li>• ECKERT, Claudia, 2014. <i>IT-Sicherheit</i>.</li><li>• MATTHES, R. , 2003. <i>Algebra, Kryptologie und Kodierungstheorie</i>.</li></ul>
<b>Anmerkungen:</b>
Keine Anmerkungen

<b>Bachelorarbeit</b>			
<b>Modulkürzel:</b>	IB_BA	<b>SPO-Nr.:</b>	33
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	7
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	Winter- und Sommersemester
<b>Modulverantwortliche(r):</b>	Regensburger, Franz		
<b>Leistungspunkte / SWS:</b>	12 ECTS / 0 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	0 h	
	Selbststudium:	300 h	
	Gesamtaufwand:	300 h	
<b>Lehrveranstaltungen des Moduls:</b>	Bachelorarbeit		
<b>Lehrformen des Moduls:</b>	Prj - Projekt		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden		
<b>Prüfungsleistungen:</b>			
Bachelor-Abschlussarbeit			
Weitere Erläuterungen:			
Keine			
<b>Voraussetzungen gemäß SPO:</b>			
Voraussetzung für die Ausgabe des Themas der Bachelorarbeit ist gemäß APO §30 unbeschadet der Regelungen der APO THI, die erfolgreiche Ableistung des praktischen Studiensemesters.			
<b>Empfohlene Voraussetzungen:</b>			
Vorheriger Besuch des Moduls Projektmanagement, Kenntnis agiler Methoden (z.B. Scrum) für das Zeitmanagement, Beherrschung einer Software zur Erstellung wissenschaftlicher Arbeiten (z.B. LaTeX, Framemaker)			
<b>Angestrebte Lernergebnisse:</b>			
Nach der erfolgreichen Erstellung der Bachelorarbeit			
<ul style="list-style-type: none"> <li>• sind die Studierenden in der Lage, die Problemstellung (Forschungsfrage), die ihrer Abschlussarbeit zugrunde liegt, kurz und prägnant zu benennen</li> <li>• können die Studierenden die Methoden, technischen Hilfsmittel und Verfahren benennen, mit deren Hilfe sie ihre Abschlussarbeit erstellt haben</li> <li>• können die Studierenden ein Problem selbstständig und unter Einsatz wissenschaftlicher Methoden bearbeiten</li> <li>• können die Studierenden Anforderungen, alternative Lösungsvorschläge sowie die Ausarbeitung einzelner Lösungsansätze bewerten und schriftlich in einer überzeugenden und nachvollziehbaren Weise darstellen</li> <li>• haben die Studierenden gelernt, eine umfangreiche Aufgabenstellung durch effektives Zeitmanagement in einem vorgegebenen Zeitrahmen zum Abschluss zu bringen</li> </ul>			

<ul style="list-style-type: none"><li>• sind die Studierenden in der Lage, die wesentlichen Inhalte ihrer Abschlussarbeit vor einem Fachpublikum zu präsentieren</li></ul>
<b>Inhalt:</b>
<ul style="list-style-type: none"><li>• Formulierung einer Forschungsfrage</li><li>• Techniken, Methoden und Vorgehensweisen des wissenschaftlichen Arbeitens</li><li>• Problemlösung unter Einsatz wissenschaftlicher Methoden</li><li>• Darstellung von Anforderungen und Lösungsansätzen</li><li>• effektives Zeitmanagement</li><li>• Inhalte vor Fachpublikum präsentieren</li></ul>
<b>Literatur:</b>
<ul style="list-style-type: none"><li>• SCHLOSSER, Joachim, 2016. <i>Wissenschaftliche Arbeiten schreiben mit LaTeX</i>. 6. Auflage. ISBN 978-3-95845-289-3</li></ul>
<b>Anmerkungen:</b>
<p>Wichtiger Hinweis: Setzen Sie Ihre Betreuer und Ihren Erstprüfer regelmäßig in Kenntnis von Ihren Fortschritten. Klären Sie insbesondere deren Erwartungen an den Inhalt der Arbeit ab.</p> <p>Dual Studierende müssen die Abschlussarbeit gemäß §30 (5) bei der Dual-Partnerfirma ableisten.</p>

<b>Seminar Bachelorarbeit</b>			
<b>Modulkürzel:</b>	IB_SBA	<b>SPO-Nr.:</b>	34
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	7
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	Winter- und Sommersemester
<b>Modulverantwortliche(r):</b>	Regensburger, Franz		
<b>Leistungspunkte / SWS:</b>	3 ECTS / 2 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	24 h	
	Selbststudium:	51 h	
	Gesamtaufwand:	75 h	
<b>Lehrveranstaltungen des Moduls:</b>	Seminar Bachelorarbeit		
<b>Lehrformen des Moduls:</b>	S - Seminar		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden		
<b>Prüfungsleistungen:</b>			
LN - Präsentation (15-30)			
Weitere Erläuterungen: Erfolgreiche Absolvierung des Online-Tests im Moodle-Kurs 'Seminar Bachelorarbeit' <a href="https://moodle.thi.de/course/view.php?id=8215">https://moodle.thi.de/course/view.php?id=8215</a>			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden			
<b>Empfohlene Voraussetzungen:</b>			
Vorheriger Besuch des Moduls Projektmanagement, Kenntnis agiler Methoden (z.B. Scrum) für das Zeitmanagement			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch des Moduls			
<ul style="list-style-type: none"> <li>kennen die Studierenden sowohl formale als auch inhaltliche Anforderungen, die an eine Bachelorarbeit gestellt werden</li> <li>sind die Studierenden mit den grundlegenden wissenschaftlichen Arbeitsmethoden vertraut, die im Rahmen der Erstellung einer Abschlussarbeit zur Anwendung kommen sollen</li> <li>haben Studierende ein besseres Verständnis darüber, wie eine Abschlussarbeit aufgebaut ist, wie Zielsetzungen/Hypothesen zu definieren und Erkenntnisse aus der Bearbeitung zu präsentieren sind</li> <li>wissen Studierende, wie man eine umfangreiche wissenschaftliche Arbeit strukturiert und prägnant einem breiten Publikum vermitteln kann</li> <li>sind Studierende darin geübt, sachlich und objektiv zu argumentieren und mit konstruktiver Kritik umzugehen</li> </ul>			

<ul style="list-style-type: none"><li>• ist den Studierenden bewusst, dass für die erfolgreiche Erstellung einer Abschlussarbeit ein explizites Zeitmanagement erforderlich ist</li></ul>
<b>Inhalt:</b>
<ul style="list-style-type: none"><li>• Individuelle Themenfindung</li><li>• Problemstellungen der betrieblichen Praxis</li><li>• Wahl des Erstprüfers, Wahl des Zweitprüfers</li><li>• Kommunikation mit dem Erstprüfer</li><li>• Einarbeitung (Inhalte, Zeitplan, Gliederung)</li><li>• Methoden des wissenschaftlichen Arbeitens: Recherche, Nachvollziehbarkeit und Reproduzierbarkeit, korrektes Zitieren und plagiatsfreies Arbeiten</li><li>• Zeitmanagement</li></ul>
<b>Literatur:</b>
<ul style="list-style-type: none"><li>• Ohne Autor. <i>Seminar Bachelorarbeit (Moodle Kurs)</i> [online]. [Zugriff am: 25.09.2023]. Verfügbar unter: <a href="https://moodle.thi.de/course/view.php?id=8215">https://moodle.thi.de/course/view.php?id=8215</a></li></ul>
<b>Anmerkungen:</b>
Keine Anmerkungen

<b>Grundlagen der Betriebswirtschaft und des Gründertums</b>			
<b>Modulkürzel:</b>	INF-B-BWLGT	<b>SPO-Nr.:</b>	35
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Pflichtfach	6
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Sommersemester
<b>Modulverantwortliche(r):</b>	Regensburger, Franz		
<b>Leistungspunkte / SWS:</b>	5 ECTS / 4 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	47 h	
	Selbststudium:	78 h	
	Gesamtaufwand:	125 h	
<b>Lehrveranstaltungen des Moduls:</b>	Grundlagen der Betriebswirtschaft und des Gründertums		
<b>Lehrformen des Moduls:</b>	SU/Ü - seminaristischer Unterricht / Übungen		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
LN - schriftliche Prüfung, 90 Minuten			
Weitere Erläuterungen: Keine			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Praktische Erfahrungen in der Realisierung von Prototypen sowie Basiskenntnisse in Statistik für die Versuchsdatenauswertung wären von Vorteil.			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch des Moduls sind die Studierenden in der Lage,			
<ul style="list-style-type: none"> <li>• die wesentlichen Merkmale unternehmensverantwortlichen Handelns zu beschreiben</li> <li>• Grundlagen der Globalisierung und der Marktwirtschaft zu verstehen</li> <li>• Marktformen und Wirtschaftsräume sowie Absatzpolitik und Marketing Mix zu unterscheiden</li> <li>• Unternehmensorganisation und Unternehmensstrukturen zu beschreiben</li> <li>• die wesentlichen Merkmale des und Vorgehensweisen im Innovationsmanagement zu beschreiben</li> <li>• Unterschiedliche Führungsstile zu benennen und verschiedene Ausprägungen der Personalorganisation zu erklären</li> <li>• die wesentlichen Aspekte des Gründertums wie Grundkenntnisse der Finanzierung, der Buchhaltung und der Investitionsrechnung zu verstehen und im praxisbezogenen Kontext anzuwenden</li> <li>• Grundlegende Formen der Material- und Produktionswirtschaft zu benennen</li> </ul>			

**Inhalt:**

Betriebswirtschaftliche Grundlagen des Gründertums:

- Grundbegriffe (Ziele, konstitutive Entscheidungen wie z.B. über Rechtsform sowie Kooperationen, Entscheidungsregeln)
- Grundlagen der Globalisierung und der Marktwirtschaft
- Marktformen und Wirtschaftsräume, Absatzpolitik und Marketing Mix
- Unternehmensorganisation, Unternehmensstrukturen
- Führungsstile und Personalorganisation
- Grundlagen der Material- und Produktionswirtschaft
- Grundkenntnisse der Finanzierung, der Buchhaltung und der Investitionsrechnung
- Innovationsmanagement (Merkmale und Vorgehensweisen)

Grundlagen Entrepreneurship und Intrapreneurship:

- Entrepreneur / Intrapreneur - Grundlagen
- Konzeptionelle Aspekte – Businessplan, Business Model Canvas, Entrepreneur Marketing, Unternehmenskultur
- Kooperationen – Inkubatoren, Akzeleratoren, Company Builder
- Gründungsfinanzierung

**Literatur:**

- THOMMEN, J.-P. und ET AL., 2020. *Allgemeine Betriebswirtschaftslehre*. 9. Auflage. Wiesbaden: Springer. ISBN <https://doi.org/10.1007/978-3-658-27246-3>
- VAHS, D. und A. BREM, 2015. *Innovationsmanagement*. 5. Auflage. Stuttgart: Schäffer-Poeschel. ISBN 978-3-7910-3420-1
- BAYSTARTUP.DE, . *Handbuch: Businessplan-Erstellung* [online]. [Zugriff am: ]. Verfügbar unter: <https://www.baystartup.de/bayerische-businessplan-wettbewerbe/handbuch-businessplan/>
- HWK-MUENCHEN.DE, . *Infomaterial Businessplan/Unternehmenskonzept, Planungs-, Formulierungshilfen, Beispiele* [online]. [Zugriff am: 16.01.2021]. Verfügbar unter: <https://www.hwk-muenchen.de/artikel/existenzgruendungsberatung-74,0,7268.html>

**Anmerkungen:**

Keine Anmerkungen



## 4.2 Fachwissenschaftliche WPM

<b>Eingebettete Systeme und Echtzeitsysteme</b>			
<b>Modulkürzel:</b>	FFI_ES	<b>SPO-Nr.:</b>	31
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Fachwissenschaftliche WPM	6
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Sommersemester
<b>Modulverantwortliche(r):</b>	Margull, Ulrich		
<b>Leistungspunkte / SWS:</b>	5 ECTS / 4 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	47 h	
	Selbststudium:	78 h	
	Gesamtaufwand:	125 h	
<b>Lehrveranstaltungen des Moduls:</b>	Eingebettete Systeme und Echtzeitsysteme		
<b>Lehrformen des Moduls:</b>	SU/Ü - seminaristischer Unterricht/Übung		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden		
<b>Prüfungsleistungen:</b>			
LN - schrP90 - schriftliche Prüfung, 90 Minuten			
Weitere Erläuterungen: Keine			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Vorheriger Besuch eines Moduls mit Inhalten zu Betriebssystemen			
<b>Angestrebte Lernergebnisse:</b>			
Nach erfolgreicher Teilnahme an den Modulveranstaltungen sind die Studierenden in der Lage,			
<ul style="list-style-type: none"> <li>• spezifische Merkmale und Anforderungen an eingebettete Systeme zu kennen und zu bewerten</li> <li>• Hardware für eingebettete Systeme zu kennen, zu vergleichen und zu bewerten</li> <li>• Schedulingverfahren für eingebettete Echtzeitsysteme wie Fixed-Priority, Earliest Deadline First, Time-Triggered, PFair) auf Single- und Multicore-Architekturen für unabhängige und für abhängige Tasks zu kennen, als Teil eingebetteter Software zu implementieren und Inhalte zu verifizieren</li> <li>• Probleme und Lösungen beim Zugriff auf Ressourcen in eingebetteten Systemen zu kennen und anzuwenden</li> <li>• Echtzeitbetriebssysteme wie VxWorks, Embedded Linux, OSEK-OS/AUTOSAR-OS zu kennen und zu bewerten</li> </ul>			

<ul style="list-style-type: none"><li>• Software als Teil einer Systemlösung unter Berücksichtigung spezifischer Anforderungen an eingebettete Systeme in Flugzeugen und in Fahrzeugen zu entwerfen, zu implementieren und zu verifizieren</li></ul>
<b>Inhalt:</b>
<ul style="list-style-type: none"><li>• Einführung: Definitionen und Beispiele, Merkmale eingebetteter Systeme bzw. cyber-physikalischer Systeme (CPS)</li><li>• Hardware eingebetteter Systeme: Übersicht, Sensoren und Aktoren, Prozessoren, Speicher, Ein-/Ausgabe und Kommunikation</li><li>• Scheduling: Grundbegriffe, Scheduling-Algorithmen für Echtzeitsysteme (EDF, RMS, DMS, TS), Abhängige Tasks, , Ressourcenzugriffe</li><li>• Multicore-Systeme: Architektur, Global-EDF, PFAIR</li><li>• Echtzeitbetriebssysteme: Überblick, OSEK-OS, AUTOSAR-OS, VxWorks, Embedded Linux</li><li>• Programmierung eingebetteter Systeme: Imperative Sprachen, Objektorientierung</li></ul>
<b>Literatur:</b>
<ul style="list-style-type: none"><li>• BUTTAZZO, Giorgio C., 2011. <i>Hard real-time computing systems: predictable scheduling algorithms and applications</i> [online]. New York, NY [u.a.]: Springer PDF e-Book. ISBN 978-1-4614-0676-1. Verfügbar unter: <a href="https://doi.org/10.1007/978-1-4614-0676-1">https://doi.org/10.1007/978-1-4614-0676-1</a>.</li><li>• MARWEDEL, Peter, 2021. <i>Eingebettete Systeme: Grundlagen eingebetteter Systeme in cyber-physikalischen Systemen</i> [online]. Wiesbaden, Germany: Springer Vieweg PDF e-Book. ISBN 978-3-658-33437-6. Verfügbar unter: <a href="https://doi.org/10.1007/978-3-658-33437-6">https://doi.org/10.1007/978-3-658-33437-6</a>.</li><li>• LEE, Edward Ashford und Sanjit Arunkumar SESHIA, 2015. <i>Introduction to embedded systems: a cyber-physical systems approach</i>. Second edition, printing 2.0. Auflage. Cambridge, Massachusetts: MIT Press. ISBN 978-0-262-53381-2</li><li>• MARWEDEL, Peter, 2021. <i>Embedded system design: embedded systems foundations of cyber-physical systems, and the Internet of Things</i> [online]. Cham, Switzerland: Springer PDF e-Book. ISBN 978-3-030-60909-2, 978-3-030-60910-8. Verfügbar unter: <a href="https://doi.org/10.1007/978-3-030-60910-8">https://doi.org/10.1007/978-3-030-60910-8</a>.</li></ul>
<b>Anmerkungen:</b>
<p>Bonuspunkteregelung: Für diese Vorlesung werden Bonuspunkte gemäß APO §25 Absatz (3) für praktische Arbeiten vergeben. Die Bonuspunkte betragen maximal 5% der in der Klausur vergebenen Punkte. Die genauen Bedingungen sind im Moodle-Kursraum zur Veranstaltung hinterlegt.</p> <p>(Link: <a href="https://moodle.thi.de/mod/resource/view.php?id=294357">https://moodle.thi.de/mod/resource/view.php?id=294357</a>)</p>

<b>Parallel Programming</b>			
<b>Modulkürzel:</b>	KI_FW_PP	<b>SPO-Nr.:</b>	31
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Fachwissenschaftliche WPM	6
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Deutsch	1 Semester	nur Wintersemester
<b>Modulverantwortliche(r):</b>	Membarth, Richard		
<b>Leistungspunkte / SWS:</b>	5 ECTS / 4 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	47 h	
	Selbststudium:	78 h	
	Gesamtaufwand:	125 h	
<b>Lehrveranstaltungen des Moduls:</b>	Parallel Programming		
<b>Lehrformen des Moduls:</b>	SU/Ü - seminaristischer Unterricht/Übung		
<b>Verwendbarkeit für andere Studiengänge:</b>	Die Verwendbarkeit für andere Studiengänge kann der Fächeranerkennungsliste der Fakultät I entnommen werden.		
<b>Prüfungsleistungen:</b>			
LN - schriftliche Prüfung, 90 Minuten			
Weitere Erläuterungen: Keine			
<b>Voraussetzungen gemäß SPO:</b>			
Dieses Fach kann nur belegt werden, wenn der zweite Studienabschnitt erreicht wurde. Dazu müssen mindestens 42 ECTS-Punkte aus dem ersten Studienabschnitt nachgewiesen werden.			
<b>Empfohlene Voraussetzungen:</b>			
Kenntnisse in C- oder C++-Programmierung oder einer anderen Systemprogrammiersprache			
<b>Angestrebte Lernergebnisse:</b>			
Nach dem Besuch des Moduls sind die Studierenden in der Lage			
<ul style="list-style-type: none"> <li>• den Aufbau und die Architektur von aktuellen Single-Core und Multi-Core Prozessoren zu verstehen</li> <li>• Mutli-Core Systeme mittels OpenMP zu programmieren</li> <li>• Anwendungen und Algorithmen auf die parallelen Berechnungseinheiten abzubilden</li> <li>• Konflikte bei der parallelen Programmierung zu erkennen und zu beheben</li> <li>• Standardtechniken zur Synchronisation von Prozessen und Threads gezielt auf Probleme der parallelen Programmierung anzuwenden</li> <li>• Anwendungen für ein gegebenes System zu optimieren.</li> </ul>			
<b>Inhalt:</b>			
<ul style="list-style-type: none"> <li>• Rechnerarchitekturen und Rechenmodelle paralleler Computer</li> <li>• Amdahlsches Gesetz und Gustafsons Gesetz</li> <li>• Formen von Parallelismus</li> </ul>			

- Synchronisationsmechanismen
- Race Conditions und Deadlocks
- Parallele Design Patterns
- Multithreaded Programmierung
- Parallele Algorithmen
- GPU Programmierung mit OpenCL
- GPU Algorithmen

**Literatur:**

- SCHMIDT, Bertil und andere, 2017. *Parallel Programming: Concepts and Practice*. Cambridge, United States: Morgan Kaufmann. ISBN 978-0-12-849890-3
- MCCOOL, Michael, James REINDERS und Arch ROBISON, 2012. *Structured Parallel Programming: Patterns for Efficient Computation*. F. Auflage. Amsterdam: Morgan Kaufmann. ISBN 978-0-12-391443-9, 0-12-391443-4
- WILLIAMS, Anthony, 2019. *C++ Concurrency in Action*. S. Auflage. Shelter Island: Manning. ISBN 978-1-61729-469-3, 1-61729-469-1
- GLEIM, Urs und Tobias SCHÜLE, 2011. *Multicore-Software: Grundlagen, Architektur und Implementierung in C/C++, Java und C#*. Heidelberg: dpunkt.verlag. ISBN 978-3-89864-758-8, 978-3-86491-080-7

**Anmerkungen:**

Keine Anmerkungen

<b>Principles of Modern Software Development</b>			
<b>Modulkürzel:</b>	IB_PMSD	<b>SPO-Nr.:</b>	31
<b>Zuordnung zum Curriculum:</b>	<b>Studiengang u. -richtung</b>	<b>Art des Moduls</b>	<b>Studiensemester</b>
	Informatik (SPO WS 18/19)	Fachwissenschaftliches Wahlpflichtfach	7
<b>Modulattribute:</b>	<b>Unterrichtssprache</b>	<b>Moduldauer</b>	<b>Angebotshäufigkeit</b>
	Englisch	1 semester	only winter term
<b>Modulverantwortliche(r):</b>	Regensburger, Franz		
<b>Leistungspunkte / SWS:</b>	5 ECTS / 4 SWS		
<b>Arbeitsaufwand:</b>	Kontaktstunden:	47 h	
	Selbststudium:	78 h	
	Gesamtaufwand:	125 h	
<b>Lehrveranstaltungen des Moduls:</b>	Principles of Modern Software Development		
<b>Lehrformen des Moduls:</b>	SU/Ü - integrated lecture/exercises		
<b>Verwendbarkeit für andere Studiengänge:</b>	The applicability for other study programs can be found in the subject recognition list of Faculty I.		
<b>Prüfungsleistungen:</b>			
LN - schriftliche Prüfung, 90 Minuten			
Weitere Erläuterungen: None			
<b>Voraussetzungen gemäß SPO:</b>			
This subject can only be taken if the second stage of the study program has been reached. To do this, at least 42 ECTS points from the first part of the study program must be proven.			
<b>Empfohlene Voraussetzungen:</b>			
Basics of programming and knowledge of at least one specific programming language			
<b>Angestrebte Lernergebnisse:</b>			
On successful completion of the course, students will be able to:			
<ul style="list-style-type: none"> <li>• understand modern software development principles</li> <li>• identify pros and cons of programming languages for specific projects</li> <li>• understand how safety and resource efficient programming can work</li> <li>• know how to apply the Rust programming language to support the goals of modern software development with focus on resource efficiency and safety</li> <li>• understand Software Testing and the respective process</li> <li>• apply techniques for profiling and tracing</li> </ul>			
<b>Inhalt:</b>			
<ul style="list-style-type: none"> <li>• Modern software development principles</li> </ul>			

- Introduction of different approaches for programming languages and the corresponding pros and cons
- Introduction and identification of safety-critical aspects in modern software
- Programming in Rust to support safe and resource-efficient programming “by design”
- Embedding of Rust in existing software development processes and interaction with other programming languages
- Techniques for profiling and tracing

**Literatur:**

- KLABNIK, Steve und Carol NICHOLS, 2018. *The Rust programming language*. San Francisco: No Starch Press. ISBN 978-1-59327-828-1, 1-59327-828-4
- MARTIN, Robert C. und andere, 2018. *Clean architecture: a craftsman's guide to software structure and design*. Boston: Prentice Hall. ISBN 978-0-13-449416-6, 0-13-449416-4
- ANDRIST, Björn, Viktor SEHR und B. GARNEY, 2020. *C++ high performance: master the art of optimizing the functioning of your C++ code*. S. Auflage. Birmingham ; Mumbai: Packt Publishing. ISBN 978-1-83921-258-1

**Anmerkungen:**

None