



Studiengangsprüfungsordnung  
für den Bachelorstudiengang  
Informatik  
an der Fachhochschule Bielefeld

**Studiengangsprüfungsordnung für den Bachelorstudiengang  
Informatik  
an der Fachhochschule Bielefeld  
(University of Applied Sciences)  
vom 20. September 2018 in der Fassung der Änderung  
vom 05. Juni 2019, 06. August 2021 und 21. September 2021**

Aufgrund des § 22 Abs. 1 Nr. 3, § 2 Abs. 4 und des § 64 Abs. 1 des Gesetzes über die Hochschulen des Landes Nordrhein-Westfalen (Hochschulgesetz – HG) vom 16. September 2014 (GV. NRW. S. 547) zuletzt geändert durch Gesetz vom 25. März 2021 (GV. NRW. S.331) hat die Fachhochschule Bielefeld in Verbindung mit der Rahmenprüfungsordnung für die Bachelorstudiengänge an der Fachhochschule Bielefeld vom 11.12.2015 (Verkündungsblatt der Fachhochschule Bielefeld – Amtliche Bekanntmachungen – 2016, Nr. 1, S. 5-25) hat der Fachbereich Campus Minden die folgende Studiengangsprüfungsordnung (SPO) erlassen:

§ 1 [zu § 1 RPO-BA] Geltungsbereich der Prüfungsordnung .....	4 -
§ 2 [zu § 3 RPO-BA] Ziel des Studiums, Akademischer Grad .....	4 -
§ 3 [zu § 5 RPO-BA] Studienbeginn, Regelstudienzeit, Semesterstruktur, Studienumfang .....	4 -
§ 4 [zu § 7 RPO-BA] Lehrformen der Module.....	4 -
§ 5 [zu § 9 RPO-BA] .....	5 -
Organisation der Prüfungen, Prüfungsorgane.....	5 -
§ 6 [zu § 10 RPO-BA] .....	5 -
Prüfende und Beisitzende .....	5 -
§ 7 [zu § 12 RPO-BA] Wiederholung von Prüfungsleistungen .....	6 -
§ 8 [zu § 14 RPO-BA] Ziel, Umfang und Form der Modulprüfungen .....	6 -
§ 9 [zu § 20 RPO-BA] Hausarbeit .....	6 -
§ 10 [zu § 21 RPO-BA] Projektarbeiten .....	7 -
§ 11 [zu § 22 RPO-BA] Performanzprüfungen .....	7 -
§ 12 Wissenschaftliches Poster.....	7 -
§ 13 Kurzpublikationsmanuskript .....	7 -
§ 14 Forschungsförderungsantrag .....	7 -
§ 15 Praktikums-, Exkursions- oder Tagesprotokoll.....	8 -
§ 16 Portfolio .....	8 -
§ 17 Lerntagebuch .....	8 -
§ 18 Parcourprüfung (Stationenprüfung) .....	9 -
§ 19 [zu § 23 RPO_BA] Bewertung von Prüfungsleistungen .....	9 -
§ 20 [zu § 15 RPO-BA] Zulassung zu Modulprüfungen (Fortschrittsregelung) .	9 -
§ 21 [zu § 24 RPO-BA] Praxisphase .....	9 -

§ 22 [zu § 25 RPO-BA] Auslandssemester .....	- 10 -
§ 23 [zu § 26 RPO-BA] Bachelorarbeit .....	- 10 -
§ 24 [zu § 27 RPO-BA] Zulassung zur Bachelorarbeit .....	- 10 -
§ 25 [zu § 28 RPO-BA] Ausgabe und Bearbeitung der Bachelorarbeit .....	- 10 -
§ 26 [zu § 29 RPO-BA] Abgabe der Bachelorarbeit .....	- 11 -
§ 27 [zu § 31 RPO-BA] Ergebnis der Bachelorprüfung .....	- 11 -
§ 28 [zu § 32 RPO-BA] Zeugnis, Gesamtnote, Bachelorurkunde, Diploma Supplement .....	- 11 -
§ 29 [zu § 33 RPO-BA] Einsicht in die Prüfungsakte .....	- 11 -
§ 30 [zu § 35 RPO-BA] Inkrafttreten, Veröffentlichung .....	- 11 -

## **§ 1 [zu § 1 RPO-BA<sup>1</sup>]** **Geltungsbereich der Prüfungsordnung**

Diese Prüfungsordnung gilt für den Abschluss des Studiums in dem Bachelorstudiengang Informatik an der Fachhochschule Bielefeld. Sie regelt in Ergänzung zur Rahmenprüfungsordnung für Bachelorstudiengänge an der Fachhochschule Bielefeld (im Folgenden RPO-BA) die Prüfungen, den Inhalt und den Aufbau des Studiums unter Berücksichtigung der fachlichen und hochschuldidaktischen Entwicklungen.

## **§ 2 [zu § 3 RPO-BA]** **Ziel des Studiums, Akademischer Grad**

- (1) Das Bachelorstudium gewährleistet auf der Grundlage wissenschaftlicher Erkenntnisse und Methoden und unter Beachtung der allgemeinen gesetzlichen Studienziele eine deutliche Berufsqualifizierung. Die Absolventinnen und Absolventen verfügen nach Abschluss des Studiums über Kompetenzen für die Aufnahme einer qualifikationsadäquaten beruflichen Tätigkeit.
- (2) Absolventinnen und Absolventen entwickeln im Studium Fähigkeiten und Kompetenzen zur Planung, Anwendung, Durchführung, Evaluation und Beurteilung von allgemeinen berufsbezogenen Aufgaben und Techniken der Informatik und deren Anwendungsgebieten. Sie erhalten Zugang zu Masterstudiengängen der Informatik und können an diesen erfolgreich teilnehmen.
- (3) Aufgrund der bestandenen Bachelorprüfung wird der akademische Grad „Bachelor of Science“ (B.Sc.) verliehen.

## **§3 [zu § 5 RPO-BA]** **Studienbeginn, Regelstudienzeit, Semesterstruktur, Studienumfang**

Das Studium im Bachelorstudiengang Informatik kann jeweils zum Wintersemester aufgenommen werden.

Die Regelstudienzeit beträgt sieben Semester. Entsprechend dem European Credit Transfer System (ECTS – Europäisches System zur Anrechnung von Studienleistungen) werden pro Semester 30 Credit Points (cps), insgesamt 210 Credit Points (cps) vergeben und den Modulen zugeordnet. Für den Erwerb eines Credits wird ein Arbeitsaufwand von durchschnittlich 30 Stunden zugrunde gelegt.

## **§ 4 [zu § 7 RPO-BA]** **Lehrformen der Module**

Zusätzlich zu den in der Rahmenprüfungsordnung vorgesehenen Lehrformen können angeboten werden:

---

<sup>1</sup> Diese Studiengangsprüfungsordnung (SPO-BA) ergänzt und spezifiziert die RPO-BA wo erforderlich, im Übrigen gilt die RPO-BA. Die SPO orientiert sich an der Gliederung der RPO, deren Nummerierung von Paragraphen wird in eckigen Klammern als synoptische Nummerierung mitgeführt. Es ist so leichter möglich, sich an beiden Ordnungen zu orientieren, da SPO-BA und RPO-BA mit dieser synoptischen Nummerierung zusammen wie ein Text gelesen werden können. Fehlt die synoptische Nummerierung, so gibt es keinen entsprechenden Paragraphen in der RPO-BA.

**1. Praktische Übungen (PÜ):** Systematisches Durcharbeiten von Lehrstoffen und Zusammenhängen anhand von interaktiven laborpraktischen und rechnergestützten Lehrangeboten wie Simulationen, numerischen Programmen, grafischen Visualisierungen, dynamischen Modellen etc. Anwendung an praktischen Beispielen auf Fälle aus der Praxis unter Einbeziehung von o.a. praktischen Angeboten. Die Lehrenden leiten die Veranstaltungen, stellen interaktive Angebote zur Verfügung, geben eine Einführung, stellen Aufgaben, geben Lösungshilfen und evaluieren die praktischen Tätigkeiten der Studierenden. Die Studierenden bereiten die Veranstaltung vor, arbeiten einzeln oder in Gruppen an den interaktiven praktischen Lehrangeboten, lösen Aufgaben teilweise selbständig, aber in enger Rückkopplung mit den Lehrenden und überprüfen ihre Ergebnisse anhand der interaktiven Angebote.

**2. Mentoring (M):** Systematische Begleitung einzelner Studierender oder von Studierendengruppen niedriger Semester (Mentee) durch Studierende höherer Semester (Mentor) über längere Zeit, in der Regel ein bis zwei Semester. Die Mentoren vertiefen eigene fachliche Kenntnisse und Kompetenzen durch regelmäßige Kontakte, Aufarbeiten, Übungsaufgaben, Laboraufgaben, Literaturhinweise, Korrekturen, Begleitung und Unterstützung beim Bearbeiten von Aufgaben, die sie in Absprache mit und Anleitung durch Lehrende den ihnen zugewiesenen Mentees angedeihen lassen. Sie erwerben Schlüsselkompetenzen im Bereich Lernorganisation, Systematisierung von Inhalten, Präsentation und Beratung.

**3. Exkursion (Ex):** Systematische Vorbereitung, Durchführung und Nachbereitung von fachlichen Inhalten in und am Beispiel von außerhochschulischen Lernorten. Anwendung fachlichen Wissens und Problemlösungskompetenzen anhand praktischer Anwendungen außerhalb der Hochschule.

#### **§ 5 [zu § 9 RPO-BA]**

##### **Organisation der Prüfungen, Prüfungsorgane**

Der Fachbereich hat einen Prüfungsausschuss als Prüfungsbehörde eingerichtet. Dieser ist folgendermaßen zusammengesetzt:

1. vier Mitgliedern der Professorenschaft, darunter einem vorsitzenden Mitglied und einem stellvertretend vorsitzenden Mitglied,
2. einem Mitglied der Mitarbeiterschaft in Lehre und Forschung mit Hochschulabschluss,
3. zwei Studierenden.

#### **§ 6 [zu § 10 RPO-BA]**

##### **Prüfende und Beisitzende**

- (1) Für schriftliche Prüfungsleistungen können akademische Mitarbeiterinnen und Mitarbeiter im Auftrag und unter Anleitung der Prüferin/des Prüfers Vorkorrekturen durchführen.
- (2) Testate können von akademischen Mitarbeiterinnen und Mitarbeitern im Auftrag und unter Anleitung der Prüferin/des Prüfers vergeben werden.<sup>2</sup>

---

<sup>2</sup> Siehe § 8 (2)

## **§ 7 [zu § 12 RPO-BA] Wiederholung von Prüfungsleistungen**

- (1) Die zweite Wiederholungsprüfung kann auf Antrag nach Absprache zwischen Prüfling und Prüfendem bei Zustimmung des Prüfenden in einer anderen als der vorgesehenen Prüfungsform abgehalten werden.
- (2) Modulprüfungen werden zu den Prüfungszeiträumen der Semester angeboten, in denen das Modul durchgeführt wurde. Zusätzliche Prüfungen in späteren Prüfungszeiträumen können angeboten werden.
- (3) Praxisphase und Bachelorarbeit können je einmal wiederholt werden.
- (4) Eine mindestens mit „ausreichend“ bewertete Prüfungsleistung kann nicht wiederholt werden.

## **§ 8 [zu § 14 RPO-BA] Ziel, Umfang und Form der Modulprüfungen**

- (1) Eine Modulprüfung kann aus den in der RPO-BA vorgesehen und/oder aus folgenden Prüfungsformen bestehen:
  1. einer Prüfung, in der in einer Verknüpfung zwischen praktischen und theoretischen Anteilen eine Fähigkeit aktuell entwickelt und verwirklicht wird („Performanzprüfung“),
  2. einem wissenschaftlichen Poster,
  3. einem Kurzpublikationsmanuskript,
  4. einem Forschungsförderungsantrag,
  5. einem Praktikums-, Exkursions- oder Tagesprotokoll,
  6. einem Portfolio,
  7. einem Lerntagebuch,
  8. einer Parcourprüfung (Stationenprüfung).
- (2) Die erfolgreiche aktive Teilnahme von Studierenden an Lehrveranstaltungen kann vom Prüfenden testiert werden (Testat). Testate sind formlose Bescheinigungen der erfolgreichen Bearbeitung von Inhalten der Lehrveranstaltung. Testate können als Prüfungsvorleistung verlangt werden.<sup>3</sup>
- (3) Prüfungen können in elektronischer Form abgenommen werden.
- (4) Schriftliche Ausarbeitungen sind innerhalb einer von der/dem Lehrenden festzulegenden Frist abzuliefern. Die Frist ist durch Aushang bekannt zu machen.

## **§ 9 [zu § 20 RPO-BA] Hausarbeit**

Die Hausarbeit ist eine schriftliche Ausarbeitung, die in der Regel 20 Seiten<sup>4</sup> nicht überschreitet und die im Rahmen einer Lehrveranstaltung oder in Verbindung mit einer Projektarbeit begleitend zu dieser erstellt wird. Sie kann nach Maßgabe der/des Lehrenden durch einen Fachvortrag von in der Regel 15 bis 45 Minuten Dauer ergänzt werden.

---

<sup>3</sup> Die Vergabe von Testaten darf in diesem Fall (dem HG NRW folgend) nicht an die Anwesenheit in Lehrveranstaltungen gekoppelt sein. Es muss dann möglich sein, Testate ohne Anwesenheit z.B. durch elektronische Abgabe zu erreichen. Zu Prüfungsvorleistungen siehe § 15 (4) RPO-BA.

<sup>4</sup> Wird in dieser SPO der Begriff „Seite“ benutzt, so soll eine solche etwa 300 bis 400

## **§ 10 [zu § 21 RPO-BA] Projektarbeiten**

Die Projektarbeit besteht aus einer schriftlichen Ausarbeitung, der Übergabe einer praktisch erarbeiteten Lösung (z.B. Software) und einer Präsentation.

## **§ 11 [zu § 22 RPO-BA] Performanzprüfungen**

- (1) Eine Performanzprüfung besteht aus zwei Anteilen (theoretisch und praktisch). Eine Teilleistung ist bestanden, wenn sie mindestens mit ausreichend bewertet worden ist. Der theoretische Anteil besteht aus einer Klausur oder einer mündlichen Prüfung. Der praktische Anteil besteht aus praxisnahen Aufgaben, die im Laufe des Semesters absolviert und bewertet werden.
- (2) Die Gesamtnote ergibt sich nach einem vom Prüfenden festgelegten Verhältnis der Bewertungen der beiden bestandenen Einzelleistungen. Wenn eine Teilleistung endgültig nicht bestanden ist, gilt die gesamte Leistung als endgültig nicht bestanden.
- (3) Einzelne bestandene Teilleistungen werden auf die Folgesemester übertragen.

## **§ 12 Wissenschaftliches Poster**

Die Studierenden formulieren und gestalten individuell oder in Gruppen wissenschaftliche Poster, die am Ende dem Plenum präsentiert werden. Inhalte der Poster können z.B. die Ergebnisse von vorangegangenen (eigenen) Forschungsprojekten oder Gruppenarbeiten zu forschungsnahen Themen sein. Die Studierenden lernen mit dieser Methode, die zentralen Phasen eines Forschungsprozesses übersichtlich, wissenschaftlich korrekt sowie ansprechend darzustellen. Das Poster kann nach Maßgabe der/des Lehrenden durch einen Fachvortrag von in der Regel 15 bis 45 Minuten Dauer ergänzt werden.

## **§ 13 Kurzpublikationsmanuskript**

Die Studierenden fertigen allein oder in Gruppen ein Manuskript an, das sich an inhaltlichen und Gestaltungsvorgaben realer wissenschaftlicher Publikationsplattformen (Konferenzen, Journals etc.) nach Vorgabe der Lehrenden orientiert. Ein Kurzpublikationsmanuskript soll pro Prüfling einen Umfang von maximal 5000 Wörtern haben. Es kann nach Maßgabe der/des Lehrenden durch einen Fachvortrag von in der Regel 15 bis 45 Minuten Dauer ergänzt werden.

## **§ 14 Forschungsförderungsantrag**

Die Studierenden fertigen allein oder in Gruppen einen Forschungsförderungsantrag bzw. eine Antragsskizze an. Dabei orientieren sie sich an inhaltlichen und Gestaltungsvorgaben realer wissenschaftlicher Forschungsausschreibungen nach Vorgabe der Lehrenden. Er kann nach Maßgabe der/des Lehrenden durch einen Fachvortrag von in der Regel 15 bis 45 Minuten Dauer ergänzt werden.

---

Wörter umfassen. Titelseiten, Verzeichnisse und Anhänge werden nicht mitgezählt.

## **§ 15**

### **Praktikums-, Exkursions- oder Tagesprotokoll**

Die Studierenden fertigen allein oder in Gruppen Protokolle über Lernereignisse wie (Betriebs-) Praktika, Exkursionen oder Tagungen an. Sie stellen dabei Lerninhalte dar und dokumentieren und reflektieren ihren eigenen Lernfortschritt und die Besonderheiten des Lernereignisses und des Lernorts in Bezug auf Inhalte und Lernfortschritt. Protokolle sollen pro Prüfling einen Umfang von maximal 5000 Wörtern haben. Das Protokoll kann nach Maßgabe der/des Lehrenden durch einen Fachvortrag von in der Regel 15 bis 45 Minuten Dauer ergänzt werden.

## **§ 16**

### **Portfolio**

In einem Lernportfolio sammeln Studierende nach zuvor festgelegten Kriterien im Rahmen der Lehrveranstaltung erstellte Arbeiten und Materialien und stellen diese in Zusammenhang mit ihrem eigenen Lernen schriftlich dar. Es werden so individuelle Lernprozesse dargestellt und reflektiert. Das Lernportfolio bietet dem Lehrenden in der Gestaltung viele Freiräume, als zentrales Element sollte der (selbst-)reflexive Anteil des Portfolios dabei aber immer enthalten sein. Das Portfolio kann nach Maßgabe der/des Lehrenden durch einen Fachvortrag von in der Regel 15 bis 45 Minuten Dauer ergänzt werden.

## **§ 17**

### **Lerntagebuch**

Das Schreiben eines Lerntagebuchs soll zu einem vertieften Verständnis des behandelten Stoffes und des eigenen Lernprozesses führen, indem es zu regelmäßiger Nachbearbeitung und Reflexion anregt. Dies bezieht sich auf alle Lerninhalte, die in Zusammenhang mit den in einem Modul besuchten Veranstaltungen behandelt wurden. Dabei können auch einzelne Veranstaltungen gesondert herausgehoben werden und im Rahmen dieser Veranstaltung behandelte Themen und Diskussionen intensiver reflektiert werden. Aus dieser Gesamtmenge von Lerngelegenheiten sollen diejenigen ausgewählt und expliziert werden, die subjektiv als bedeutsam, interessant oder neuartig empfunden wurden. Das Lerntagebuch soll außerdem das Bewusstsein für den eigenen Lernprozess fördern. Es dient also der Überwachung des eigenen Verstehens und unterstützt damit die Konstruktion subjektiv bedeutsamen Wissens. Die kontinuierliche Dokumentation und Reflexion der Lernerfahrungen führt zu einem besseren Verständnis des eigenen Arbeitsverhaltens und auf diese Weise zur Entwicklung individueller Lern- und Arbeitsstrategien. Die regelmäßige schriftliche Explikation der eigenen Gedanken in kompakter Form stellt aber auch außerhalb des Veranstaltungskontexts eine sinnvolle Form der Förderung von Lernprozessen dar. Die „Verschriftlichung“ der eigenen Gedanken kann insbesondere helfen, eigene Ideen zu generieren. Die Erstellung des Lerntagebuchs ist daher auch als das Einüben einer „Technik“ des aktiven, selbstgesteuerten Lernens zu sehen. Das Portfolio kann nach Maßgabe der/des Lehrenden durch einen Fachvortrag von in der Regel 15 bis 45 Minuten Dauer ergänzt werden.



## **§ 18**

### **Parcourprüfung (Stationenprüfung)**

Bei Parcourprüfungen durchlaufen Studierende individuell oder in kleinen Gruppen simultan im Rotationsverfahren eine bestimmte Anzahl an Prüfungsstationen, die das Erreichen theoretischer und praktischer Kompetenzen überprüfen. An jeder Station werden die Leistungen der zu prüfenden Studierenden dokumentiert. Diese Dokumentation kann unter der Aufsicht der Lehrenden im Peer-Review Verfahren erfolgen.

## **§ 19 [zu § 23 RPO\_BA]**

### **Bewertung von Prüfungsleistungen**

Einzelne Module können nur mit den Prädikaten „bestanden“ oder „nicht bestanden“ bewertet werden. Die Bewertung dieser Module geht nicht in die Ermittlung der Gesamtnote des Bachelorabschlusses ein.

## **§ 20 [zu § 15 RPO-BA]**

### **Zulassung zu Modulprüfungen (Fortschrittsregelung)**

Es gilt ein semesterweise aufbauender Mindest-Leistungsfortschritt wie folgt:

1. Für die Anmeldung und Zulassung zu Modulprüfungen des 4. Semesters ist eine Mindestanzahl von 50 cps erforderlich.
2. Für die Anmeldung und Zulassung zu Modulprüfungen des 5. Semesters ist eine Mindestanzahl von 80 cps erforderlich.
3. Für die Anmeldung und Zulassung zu Modulprüfungen des 6. Semesters ist eine Mindestanzahl von 110 cps erforderlich.

## **§ 21 [zu § 24 RPO-BA]**

### **Praxisphase**

- (1) Der Bachelorstudiengang Informatik beinhaltet im 7. Semester eine berufspraktische Tätigkeit von 13 Wochen, deren Arbeitsaufwand 18 ECTS-Punkte beträgt. Auf Antrag wird zur Praxisphase zugelassen, wer mindestens 110 cps erreicht hat. Über die Zulassung entscheidet der Prüfungsausschuss auf Antrag.
- (2) Die Praxisphase bedarf der Betreuung durch eine(n) Lehrende(n) des Studienganges Bachelor in Informatik, der/die Betreuende ist in der Regel auch Prüfende(r).
- (3) Die/der Prüfende bescheinigt das Bestehen der Praxisphase, wenn die Studierenden nach dem Zeugnis der Ausbildungsstätte die ihnen übertragenden Arbeiten zufriedenstellend ausgeführt haben und ein schriftlicher Bericht, der 13 Seiten Umfang nicht unterschreitet und 20 Seiten Umfang nicht überschreitet, innerhalb von 6 Wochen nach Abschluss der Praxisphase vorgelegt worden ist.
- (4) Als Praxisstelle kommen alle Einrichtungen in Betracht, deren Aufgaben den ständigen Einsatz von Mitarbeitern mit der Qualifikation des Studienganges Informatik erlaubt. Die Einrichtungen müssen über Personal verfügen, die von ihrer Qualifikation her geeignet sind, die Studierenden während der Praxisphase zu betreuen. Die Einrichtungen müssen in der Lage sein, eine dem Ziel der Praxisphase entsprechende innerbetriebliche Tätigkeit sicherzustellen.

- (5) Die Eignung einer Praxisstelle wird nach Möglichkeit von einer/einem betreuenden Lehrenden vor Antritt der Praxisphase festgestellt und dem Prüfungsausschuss mitgeteilt.

### **§ 22 [zu § 25 RPO-BA] Auslandssemester**

- (1) Die Zulassung zu Auslandssemestern erfolgt nach Einzelfallprüfung durch den Prüfungsausschuss. Hierzu legt der Studierende und ein betreuender Lehrender bzw. eine betreuende Lehrende dem Prüfungsausschuss einen formlosen Antrag vor, der ein Learning Agreement enthält.
- (2) Das Learning Agreement kann außer der konkreten Benennung von Lehrveranstaltungen auch Themenbereiche umfassen, die durch jeweils aktuelle (kurzfristig angekündigte) Veranstaltungen an der besuchten Hochschule konkretisiert werden.
- (3) Studien- und Prüfungsleistungen, die an einer ausländischen Hochschule erworben werden und die nicht inhaltlich mit Modulen des Studiengangs Bachelor in Informatik übereinstimmen, können als Wahlpflichtmodule anerkannt werden, sofern sie Inhalte des Studiengangs sinnvoll ergänzen, die gleiche Anzahl an Credit Points umfassen und in der Form den Modulen der Wahlpflichtkataloge ähnlich sind. Über die Anerkennung entscheidet auf Antrag das vorsitzende Mitglied des Prüfungsausschusses.

### **§ 23 [zu § 26 RPO-BA] Bachelorarbeit**

- (1) Die Bachelorarbeit muss zeigen, dass die/der Studierende befähigt ist, innerhalb einer vorgegebenen Frist eine praxisorientierte Aufgabe aus ihrem/seinem Fachgebiet, sowohl in ihren fachlichen Einzelheiten als auch in den fachübergreifenden Zusammenhängen nach wissenschaftlichen Methoden selbständig zu bearbeiten.
- (2) Die Bachelorarbeit umfasst die selbständige Bearbeitung einer wissenschaftlich angewandten Problemstellung, deren Dokumentation sowie im Regelfall die Konzeption und Entwicklung einer Softwarelösung. Der Umfang der Bachelorarbeit soll 30 Seiten nicht unterschreiten und 80 Seiten nicht überschreiten.
- (3) Erstprüfer können Professoren oder Lehrkräfte für besondere Aufgaben sein, die in den Studiengängen Bachelor in Informatik oder Master in Informatik des Fachbereichs Campus Minden regelmäßig lehren.

### **§ 24 [zu § 27 RPO-BA] Zulassung zur Bachelorarbeit**

Zur Bachelorarbeit wird zugelassen, wer die Modulprüfungen bis auf vier bestanden hat.

### **§ 25 [zu § 28 RPO-BA] Ausgabe und Bearbeitung der Bachelorarbeit**

Die Bearbeitungszeit (Zeitraum von der Ausgabe bis zur Abgabe der Bachelorarbeit) beträgt höchstens drei Monate.

**§ 26 [zu § 29 RPO-BA]  
Abgabe der Bachelorarbeit**

Die Bachelorarbeit ist in schriftlicher Form in zwei gebundenen Exemplaren abzuliefern. Die Prüfenden können eine zusätzliche Abgabe in elektronischer Form verlangen.

**§ 27 [zu § 31 RPO-BA]  
Ergebnis der Bachelorprüfung**

- (1) Die Bachelorprüfung ist bestanden, wenn 210 Credits erreicht werden.
- (2) Die Bachelorprüfung ist nicht bestanden, wenn
  1. die Gesamtnote nicht mindestens „ausreichend“ (4,0) ist oder
  2. die Bachelorarbeit im zweiten Versuch nicht bestanden ist oder als nicht bestanden gilt.

**§ 28 [zu § 32 RPO-BA]  
Zeugnis, Gesamtnote, Bachelorurkunde, Diploma Supplement**

Zur Ermittlung der Gesamtnote für das Bachelorstudium werden die Noten für die einzelnen benoteten Prüfungsleistungen mit den jeweiligen ausgewiesenen Credits multipliziert. Die Summe der gewichteten Noten wird anschließend durch die Gesamtzahl der einbezogenen Credits dividiert.

**§ 29 [zu § 33 RPO-BA]  
Einsicht in die Prüfungsakte**

- (1) Die Einsichtnahme in die Prüfungsakte im Sinne von § 33 BA-RPO ist binnen eines Jahres nach Aushändigung des Prüfungszeugnisses oder des Bescheides über die nicht bestandene Bachelorprüfung zu beantragen. § 32 des Verwaltungsverfahrensgesetzes für das Land Nordrhein-Westfalen über die Wiedereinsetzung in den vorigen Stand gilt entsprechend. Der Antrag ist an das Prüfungsamt zu stellen.
- (2) Die Einsichtnahme in die Prüfungsunterlagen, die sich auf eine Modulprüfung oder eine ergänzende Studienleistung beziehen, wird auf Antrag beim Prüfungsausschuss bereits nach Ablegung der jeweiligen Prüfung gestattet. Der Antrag ist binnen eines Monats nach Bekanntgabe des Prüfungsergebnisses zu stellen.

**§ 30 [zu § 35 RPO-BA]  
Inkrafttreten, Veröffentlichung**

Diese Studiengangsprüfungsordnung wird im Verkündungsblatt der Fachhochschule Bielefeld – Amtliche Bekanntmachungen – bekannt gegeben. Sie tritt einen Tag nach ihrer Veröffentlichung in Kraft.

---

Ausgefertigt aufgrund des Beschlusses des Fachbereichsrats des Fachbereichs Campus Minden der Fachhochschule Bielefeld vom 12.07.2018.

---

Bielefeld, den 20. September 2018

Die Präsidentin  
der Fachhochschule Bielefeld

Prof. Dr. Ingeborg Schramm-Wölk





## Wahlfächer Liste 1

Datenbanken II	5.10	DB2
Einführung in Audiovisual Computing	5.11	AV1
Funktionale Programmierung	5.12	FP
Cloud Computing	5.13	CC
Künstliche Intelligenz	5.14	KI
Methoden der Computergrafik, Schwerpunkt algorithmische Geometrie und geometrische Modellierung	5.15	GL1
Methoden der Computergrafik, Schwerpunkt Bildgenerierung	5.16	VR1
Praktische Aspekte der IT-Sicherheit	5.18	PIS
Sicherheit und Zuverlässigkeit	5.19	BSZ
Spezielle Methoden der Programmierung	5.20	SM

## Wahlfächer Liste 2

Anwendungen der Computergrafik, Schwerpunkt CAGD und effiziente Datenstrukturen	6.20	CG2
Anwendungen der Computergrafik, Schwerpunkt Virtual Reality und Visualisierungstechniken	6.21	VR2
Computerkunst, Sozioinformatik und neue Kommunikationsalgorithmen	6.22	CSK
Datenbankanwendungen	6.23	DBA
Embedded Software	6.24	ESW
Internetsicherheit	6.25	BIS
Mobile Applikationen	6.26	MOB
Full Stack Development	6.27	FSD
Softwarequalität	6,28	SQ
Webengineering	6.29	WE
Anwendungen der Künstlichen Intelligenz	6.30	AKI
Game Programming	6.31	GPR
Computer Vision	6.32	CV

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Algorithmen und Datenstrukturen								Kürzel ADS
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
2.2	150h	5	2. Sem.	jährlich	SoSe	1 Sem.	Pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>	<b>gepl. Gruppengr.</b>	<b>Sprache</b>		
	Vorlesung	2 SWS / 30h	45h	Wird in LV bekannt gegeben.	60	deutsch		
	Praktikum / Seminar	2 SWS / 30h	45h		15	deutsch		
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>							
	<ul style="list-style-type: none"> <li>Die Studierenden kennen das O-Kalkül und können es einsetzen zur Abschätzung des Laufzeitverhaltens von Algorithmen.</li> <li>Sie kennen Verfahren zur Algorithmenentwicklung und Strategien zur Lösung von Optimierungsproblemen und können sie auf Beispiele anwenden.</li> <li>Die Studierenden kennen Standard-Datenstrukturen, sowie mehrere Baumarten, und können sie zur Modellierung und in der Softwareentwicklung einsetzen.</li> <li>Sie kennen die Einsatzgebiete und Vor- und Nachteile verschiedener Such- und Sortierverfahren und können sie zielgerichtet in der Programmierung einsetzen.</li> <li>Sie kennen unterschiedliche Methoden des Hashens, können sie bewerten und auf Beispiele anwenden.</li> <li>Die Studierenden kennen einige Graphalgorithmen und Anwendungen davon. Sie können jeweils ein angemessenes Verfahren auswählen und implementieren. Sie wissen um die Komplexitätsklassen P und NP und deren Bedeutung.</li> </ul>							
3	<b>Inhalte</b>							
	<ul style="list-style-type: none"> <li>O-Kalkül</li> <li>Algorithmen- und Programmspezifikationen</li> <li>Algorithmenentwicklung mit schrittweiser Verfeinerung, Rekursion, Bibliotheken und Entwurfsmuster</li> <li>Algorithmen zur Lösung von Optimierungsproblemen, optimale und nicht-optimale Verfahren: dynamisches Programmieren, Backtracking, Teile-und-herrsche, Greedy-Verfahren</li> <li>Datenstrukturen: Reihung, verkettete Liste, Stack, Queue, Skipliste</li> <li>Bäume: beispielsweise Binärbäume, AVL-Bäume, 2-3-4-Bäume, Rot-Schwarz-Bäume</li> <li>Suchverfahren und Sortierverfahren: beispielsweise HeapSort, SelectionSort, InsertionSort, BubbleSort, QuickSort, MergeSort</li> <li>Hashverfahren</li> <li>Graphalgorithmen: Breitensuche, Tiefensuche, Topologisches Sortieren</li> <li>Komplexitätsklassen P und NP</li> </ul>							
4	<b>Teilnahmevoraussetzungen</b>							
	keine							
5	<b>Prüfungsgestaltung</b>							
	Performanzprüfung oder Klausur oder Hausarbeit oder Parcours-prüfung (Stationenprüfung)							
6	<b>Voraussetzung für die Vergabe von Credit Points</b>							
	Bestandene Modulprüfung							
7	<b>Verwendung des Moduls (in folgenden Studiengängen):</b>							
	Bachelorstudiengang Informatik							
8	<b>Modulbeauftragte/r</b>							
	Dipl.-Inf. B.C. George							
9	<b>Sonstige Informationen</b>							



**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Betriebssysteme								Kürzel BES
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
4.1	150h	5	4. Sem.	jährlich	SoSe	1 Sem.	Pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>	
	Vorlesung	2 SWS / 30h	45h	Wird in LV bekannt gegeben.		60	deutsch	
	Praktikum / Seminar	2 SWS / 30h	45h			15	deutsch	
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>							
<p>Betriebssysteme verwalten die Ressourcen eines Rechnersystems und bestimmen dadurch wesentlich dessen Leistungsfähigkeit und Nutzbarkeit. Gute Kenntnisse der Betriebssystem-Konzepte sind daher für das Verständnis moderner IT Systeme unerlässlich. Studierende besitzen nach Abschluss der Lehrveranstaltung die folgenden Fähigkeiten:</p> <ul style="list-style-type: none"> <li>- Systembezogene Abstraktionen kennen, verstehen und wiedererkennen, wie sie von Betriebssystemen verwendet und bereitgestellt werden</li> <li>- Wichtige Verfahren und Algorithmen aus dem Bereich Betriebssysteme sicher anwenden</li> <li>- Methoden kennen und verstehen, die bei der Konstruktion von Betriebssystemen Verwendung finden</li> <li>- beides an Beispielsystemen detailliert erläutern können (primär UNIX/Linux, z.T. Windows und andere aktuelle Betriebssysteme)</li> <li>- Die Studierenden können im Team hardwarenahe Software (z.B. Linux Kerneltreiber) erstellen und eine Argumentation/Strategie entwerfen, um Designentscheidungen zu begründen.</li> </ul>								
3	<b>Inhalte</b>							
<ul style="list-style-type: none"> <li>- Systemarchitektur moderner Rechensysteme</li> <li>- Aufgaben, Arten und Strukturen von Betriebssystemen</li> <li>- ggf. Kurzvorstellung aktuelle Rechnerarchitekturen (Pipeline, Cache, Branch Prediction)</li> <li>- Nebenläufigkeit, Prozessverwaltung, Scheduling</li> <li>- Synchronisation und Kommunikation</li> <li>- Speicherverwaltung: Virtueller Speicher, Segmentierung, Paging, Seitenersetzungsalgorithmen</li> <li>- Ein-/Ausgabe: zeichen- und blockorientierte Geräte</li> <li>- Dateisysteme: Beispiele, Aufbau, Fehlertoleranz</li> <li>- Treiber-Modelle und -Programmierung</li> <li>- Sicherheit: Authentifikation, Schutzmechanismen, Autorisierung, vertrauenswürdige Systeme</li> <li>- Wechselnde Inhalte der Praktika zu aktuellen Themen (z.B. Bash-Skripting, Entwicklung einer Shell, Entwicklung eines Linux Kerneltreibers)</li> </ul>								
4	<b>Teilnahmevoraussetzungen</b>							
Formal: -, Inhaltlich: Kenntnisse aus dem Modul 3.3 Systemprogrammierung (SP)								
5	<b>Prüfungsgestaltung</b>							
Performanzprüfung								
6	<b>Voraussetzung für die Vergabe von Credit Points</b>							
Bestandene Modulprüfung								
7	<b>Verwendung des Moduls</b> (in folgenden Studiengängen):							
Bachelor Informatik								
8	<b>Modulbeauftragte/r</b>							
Prof. Dr.-Ing. Martin Hoffmann								
9	<b>Sonstige Informationen</b>							
Literatur: - Tanenbaum: "Moderne Betriebssysteme", Pearson Studium, 2016 - Stallings: "Operating Systems: Internals and Design Principles", Prentice Hall, 2011								

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

<b>Bildverarbeitung und Mustererkennung</b>								<b>Kürzel BVM</b>
<b>Nr.</b>	<b>Workload</b>	<b>Credit Points</b>	<b>Studien- semester</b>	<b>Häufigkeit</b>	<b>Sem.</b>	<b>Dauer</b>	<b>Art</b>	<b>Q-Niveau</b>
<b>4.2</b>	<b>150h</b>	<b>5</b>	4. Sem	jährlich	SoSe	1 Sem	Pflicht	B.Sc.
<b>1a</b>	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>	<b>gepl. Gruppengr.</b>	<b>Sprache</b>		
	Vorlesung	2 SWS / 30h	45h	Wird in LV bekannt gegeben	60	deutsch		
	Praktikum	2 SWS / 30h	45h		15	deutsch		
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>							
	Die Studierenden kennen und verstehen die grundlegenden Methoden und Verfahren der Bildverarbeitung und Mustererkennung. Sie sind in der Lage diese auf typische Problemstellungen zu übertragen und geeignete Lösungsvorschläge zu erarbeiten und zu bewerten. Im Rahmen des Praktikums erlernen die Studierenden darüber hinaus eigene Programme innerhalb einer vorgegebenen Frist selbständig zu entwickeln. Sie setzen grundlegende Algorithmen prototypisch um und evaluieren diese anhand von Beispieldaten. Dabei setzen sie fachspezifische Softwarebibliotheken wie OpenCV ein.							
<b>3</b>	<b>Inhalte</b>							
	Bildverarbeitung und Mustererkennung sind interdisziplinäre Arbeitsgebiete, in denen Methoden zur Analyse und Interpretation von Einzelbildern und Bildfolgen genutzt werden. Anwendungen finden sich beispielsweise in der Sicherheitstechnik, Fernerkundung, Maschinenbau, oder der medizinischen Diagnoseunterstützung. Folgende Themengebiete stehen beispielhaft für mögliche Inhalte der Veranstaltung: Farbräume, Bildverbesserung und Filter, Segmentierung, Merkmalsextraktion, Klassifikationsverfahren, Clustering, Deep Learning.							
<b>4</b>	<b>Teilnahmevoraussetzungen</b>							
	Keine							
<b>5</b>	<b>Prüfungsgestaltung</b>							
	Klausur oder mündliche Prüfung oder schriftliche Hausarbeit oder Projektarbeit oder Performanzprüfung oder wissenschaftliches Poster oder Kurzpublikationsmanuskript oder Forschungsförderungsantrag oder Praktikums-, Exkursions- oder Tagesprotokoll oder Portfolio oder Lerntagebuch oder Parcourprüfung oder (gem. § 14 (4) RPO) eine Kombination aus verschiedenen Formen von Prüfungsleistungen. Prüfungen können in digitaler Form abgenommen werden.							
<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b>							
	Testat und bestandene Modulprüfung							
<b>7</b>	<b>Verwendung des Moduls</b> (in folgenden Studiengängen):							
	Bachelorstudiengang Informatik							
<b>8</b>	<b>Modulbeauftragte/r</b>							
	Prof. Dr.-Ing. Jan Rexilius							
<b>9</b>	<b>Sonstige Informationen</b>							
	Literatur wird in der Veranstaltung bekannt gegeben.							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Computergrafik								Kürzel CG
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
4.3	150 h	5	4 Sem.	jährlich	SoSe	1 Sem.	Pflicht	B.Sc.
	Vorlesung		2 SWS / 30 h	45 h	Wird in LV bekannt gegeben.		60	deutsch
	Praktikum		2 SWS/ 30 h	45 h			15	
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Die Studierenden können die fundamentalen Methoden und Algorithmen in der Computergrafik anwenden. Die Studierenden sind in der Lage grundlegende Begriffe zu benennen und sowohl Verfahren aus der 2D wie auch 3D Computergrafik zu erläutern. Die mathematischen Grundlagen für die 2D und 3D Computergrafik können sie zusammenfassen. Die Studierenden können gängige Tools und Anwender-Werkzeuge aus der Computergrafik benutzen und sind mit den zugehörigen Technologien vertraut.							
3	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Historie, Überblick, Beispiele: Anwender-Werkzeuge, Tools, Anwendungsbeispiele.</li> <li>• Begriffe und Grundlagen: Grafische Eingabegeräte, Bildschirmtechnologien, 3D-Sichtsysteme, Rastergrafik.</li> <li>• Objekt- und Sichttransformationen: Koordinatensysteme, Transformationen, Projektionen, Clipping.</li> <li>• Repräsentation und Modellierung von Objekten: Polygonale Repräsentation, Raumteilungsverfahren, Szenenbeschreibung.</li> <li>• Rendering und Visibilität: Farbmodelle, Visibilitätsverfahren, Beleuchtung und Schattierung, lokale Beleuchtungsmodelle, interpolative Schattierungstechniken, globale Beleuchtungsmodelle, Rendering-Pipelines.</li> <li>• Tools zur Modellierung/Rendering: Modellierung und Rendern einer kleinen Szene mit Hilfe z.B. von Autodesk Maya, Blender.</li> </ul>							
4	<b>Teilnahmevoraussetzungen</b> Formal: - Inhaltlich: Kenntnisse aus dem Modul 1.1 Mathematik 1 (MA1) Es gilt § 17 „Fortschrittsregelung“ dieser BPO							
5	<b>Prüfungsgestaltung</b> Performanzprüfung							
6	<b>Voraussetzung für die Vergabe von Credit Points</b> Bestandene Modulprüfung							
7	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
8	<b>Modulbeauftragte/r</b> Prof. Dr. Kerstin Müller							
9	<b>Sonstige Informationen</b> <b>Literatur</b> <ul style="list-style-type: none"> <li>• Bender M., Brill, M.: Computergrafik, 2. Auflage, Hanser Verlag, 2005. <a href="http://www.vislab.de">http://www.vislab.de</a></li> <li>• Hearn D., Baker M.P.: Computer Graphics with OpenGL, Pearson International Edition.</li> <li>• Foley J., van Dam A., Feiner S., Hughes J.: Computer Graphics – Principles and Practice, Addison-Wesley</li> </ul>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Datenbanken I								Kürzel DB1
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
3.2	150h	5	3. Sem.	jährlich	WiSe	1 Sem.	Pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>	<b>gepl. Gruppengr.</b>	<b>Sprache</b>		
	Vorlesung	2 SWS / 30h	45h	Wird in LV bekannt gegeben.	60	deutsch		
	Praktikum	2 SWS / 30h	45h		15	deutsch		
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Datenbanken sind eine Basistechnologie der Informatik. Ziel der Veranstaltung ist ein sicheres Verständnis der grundlegenden Konzepte und Sprachen von Datenbanksystemen zu vermitteln. Die Studierenden wenden ein einheitliches konsistentes Begriffsgebäude bezüglich der Datenbankthematik an. Sie können die theoretischen Grundlagen von Datenbanksystemen an Beispielen erläutern, insbesondere funktionale Abhängigkeiten, relationale Algebra und Normalisierung, sie wissen um die Aufgaben und den Sinn der grundlegenden Architektur von DBMS. Sie können die Dokumentation mehrerer wichtiger DBMS gezielt problemlösend durchsuchen und kennen die grundlegenden Funktionen der Clients mehrerer DBMS und nutzen Sie zur Kommunikation und Programmierung von Datenbanken. Sie modellieren komplexe Sachverhalte sicher in auch umfangreiche Datenmodelle und implementieren sie in verschiedenen DBMS. Sie fällen dabei begründete Entscheidungen für die Anwendungen von Constraints, Domänen und Datentypen. Sie wenden SQL sicher zur Lösung komplexer Informationsbedarfe an und erstellen umfangreiche nichttriviale Abfragen. Dabei verwenden sie sowohl den aktuellen SQL Standard (zur Zeit SQL:2016) als auch die Dialekte mehrerer wichtiger DBMS. Sie verstehen den Transaktionsbegriff, beschreiben Probleme/Phänomene der Mehrbenutzersynchronisation und Nebenläufigkeit in Read/Write Notation, und entscheiden, wie man sie durch Isolation von Transaktionen verhindert – sowohl durch Standard-Isolationslevel, als auch durch spezifische Implementierungen in mehreren DBMS. Sie greifen über Datenbankschnittstellen aus eigenen Programmen auf Datenbanken zu und verarbeiten Datensätze in Programmen und Datenbanken. Sie programmieren Persistent Stored Modules in einem der besprochenen DBMS.							
3	<b>Inhalte</b> Folgende Themenbereichen stehen exemplarisch für mögliche Inhalte: <ul style="list-style-type: none"> <li>• Aufgaben und Architektur von Datenbanksystemen</li> <li>• Clients und Schnittstellen zu Datenbanksystemen</li> <li>• Grundlagen des relationalen Modells</li> <li>• E/R Modellierung, logische und physische Datenmodelle, SQL Datentypen, Implementierung in wichtigen DBMS</li> <li>• Constraints, Assertions, Integrität, Domänen, Datentypen</li> <li>• SQL:2016, insbesondere SQL-schema statements, SQL-data statements, SQL-data change statements, SQL-transaction statements und SQL-connection statements</li> <li>• Transaktionskonzepte, Nebenläufigkeit, Isolationslevel</li> <li>• Datenbankschnittstellen (JDBC, ODBC)</li> </ul> Grundlagen von Persistent Stored Modules, Programmierung von PSM, Trigger							
4	<b>Teilnahmevoraussetzungen</b> Formal: - Inhaltlich: -							
5	<b>Prüfungsgestaltung</b> Performanzprüfung oder mündliche Prüfung oder Hausarbeit oder Projektarbeit oder Praktikums-, Exkursions- oder Tagesprotokoll oder Portfolio oder Lerntagebuch oder Parcourprüfung							
6	<b>Voraussetzung für die Vergabe von Credit Points</b> Praktikum mit Testat Bestandene Modulprüfung							
7	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
8	<b>Modulbeauftragte/r</b> Prof. Dr. Dominic Becking							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

<b>9</b>	<b>Sonstige Informationen</b> <ul style="list-style-type: none"><li>• Kleuker, S., Grundkurs Datenbankentwicklung, 4. Aufl. Vieweg Teubner, 2016</li><li>• Kemper, A, Eickler, A, Datenbanksysteme - Eine Einführung, 10. Aufl. De Gruyter, 2015</li><li>• Elmasri, R. A., Navathe, B. N., Grundlagen von Datenbanksystemen, Hanser, 2009</li><li>• Piepmeyer, L., Grundkurs Datenbanksysteme, Hanser, 2011</li><li>• Saake, S., Sattler, K.-U., Heuer, A., Datenbanken - Konzepte und Sprachen, mitp, 2010</li></ul> Aktuelle Literatur zu Datenbanksystemen
----------	---

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Einführung in die Informatik								Kürzel EIN
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
1.0	150h	5	1. Sem.	jährlich	WiSe	1 Sem.	Pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>	<b>gepl. Gruppengr.</b>	<b>Sprache</b>		
	Vorlesung	2 SWS / 30h	45h	Wird in LV bekannt gegeben.	60	deutsch		
	Übung	2 SWS / 30h	45h		30	deutsch		
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Es sollen die Wurzeln und Entwicklungsgeschichte der Informatik verstanden werden. Die Studierenden sollen die wechselseitigen Einwirkungen der Gesellschaft auf die Informatik und umgekehrt erkennen und selbst lernen, dazu Position zu beziehen. Ein besonderer Aspekt ist der Gleichstellung der Geschlechter in der Informatik in Forschung, Lehre und Entwicklung gewidmet. Die Studierenden sollen vielfältige Berufsbilder von Informatikern aus verschiedenen Branchen kennen lernen. Das Fachgebiet soll in seiner Vielfalt und Breite und den begrifflichen Teilgebieten verstanden werden. Die Studierenden sollen befähigt werden in ihrem nachfolgenden Studium eine gezielte Fächerwahl zu treffen, unter Beachtung ihrer Stärken und Neigungen und im Hinblick auf ihr zukünftig angestrebtes Berufsfeld. Die Studierenden sollen in Ihrer Studienwahl bestärkt werden und für ein erfolgreiches Studium motiviert werden.							
3	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Geschichte der Informatik</li> <li>• Errungenschaften der Informatik</li> <li>• Informatik und Gesellschaft</li> <li>• Gleichstellung der Geschlechter in der Informatik</li> <li>• Teilgebiete der Informatik</li> <li>• Bedeutung des Datenschutzes</li> <li>• Berufsbilder von Informatikern</li> <li>• Speicherung von Information</li> <li>• Aussagenlogik</li> <li>• Formale Sprachen</li> <li>• Grafen</li> <li>• Algorithmen</li> <li>• Codierung</li> <li>• Künstliche Intelligenz</li> <li>• Internet</li> </ul>							
4	<b>Teilnahmevoraussetzungen</b> keine							
5	<b>Prüfungsgestaltung</b> Klausur							
6	<b>Voraussetzung für die Vergabe von Credit Points</b> Bestandene Klausur							
7	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
8	<b>Modulbeauftragte/r</b> Prof. Dr.-Ing. Grit Behrens							
9	<b>Sonstige Informationen</b>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Einführung in die Programmierung mit Skriptsprachen								Kürzel EPS
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
1.3	150h	5	1. Sem.	jährlich	WiSe	1 Sem.	Pflicht	B.Sc.
<b>1</b>	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>	<b>gepl. Gruppengr.</b>	<b>Sprache</b>		
	Vorlesung	2 SWS / 30h	45h	Wird in LV bekannt gegeben.	60	deutsch		
	Praktikum / Seminar	2 SWS / 30h	45h		15	deutsch		
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>							
	<ul style="list-style-type: none"> <li>• Die Studierenden kennen mindestens eine Skriptsprache und können sie zur Entwicklung eigener Applikationen nutzen.</li> <li>• Sie kennen Strukturelemente von imperativen Programmiersprachen und setzen sie in eigenen Programmen ein.</li> <li>• Sie können Sprachkonzepte von Skriptsprachen in eigenen Applikationen anwenden.</li> <li>• Sie können in einfachen Beispielen Daten in Bezug auf ihre Struktur analysieren und als JSON- oder XML-Dokument identifizieren.</li> <li>• Sie können JSON- oder XML-Dokumente mittels einer Skriptsprache erstellen und verarbeiten.</li> <li>• Sie verstehen verschieden Aspekte des Teamworks und können ihre Teamrollen nach Belbin identifizieren.</li> </ul>							
<b>3</b>	<b>Inhalte</b>							
	<ul style="list-style-type: none"> <li>• Algorithmenbegriff</li> <li>• Strukturelemente von Algorithmen</li> <li>• Eigenschaften von Skriptsprachen, ihre Vor- und Nachteile</li> <li>• Sprachkonzepte von Skriptsprachen</li> <li>• Aufbau von JSON- oder XML-Dokumenten</li> <li>• Standardbibliotheken zur Verarbeitung von JSON- oder XML-Dateien</li> <li>• Teamrollen nach Belbin</li> </ul>							
<b>4</b>	<b>Teilnahmevoraussetzungen</b>							
	keine							
<b>5</b>	<b>Prüfungsgestaltung</b>							
	Performanzprüfung							
<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b>							
	Bestandene Modulprüfung							
<b>7</b>	<b>Verwendung des Moduls (in folgenden Studiengängen):</b>							
	Bachelorstudiengang Informatik							
<b>8</b>	<b>Modulbeauftragte/r</b>							
	Dipl.-Inf. B.C. George							
<b>9</b>	<b>Sonstige Informationen</b>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Embedded Systems								Kürzel ES
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
3.1	150h	5	4. Sem.	jährlich	SoSe	1 Sem.	Pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>	<b>gepl. Gruppengr.</b>	<b>Sprache</b>		
	Vorlesung	2 SWS / 30h	45h	Wird in LV bekannt gegeben.	60	deutsch		
	Praktikum / Seminar	2 SWS / 30h	45h		15	deutsch		
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Die Studierenden erlernen die grundlegenden Kenntnisse zur Umsetzung von eingebetteten Systemen (embedded systems). Sie erhalten ein Verständnis der Besonderheiten des Entwurfs eingebetteter Systeme und der zur Implementierung eingebetteter Systeme benötigten Kompetenzen. Insbesondere werden den Studierenden als "Handwerkszeug" die entsprechenden Spezifikations- und Programmiertechniken, Modelle der Ablaufplanung und Software- und Systemarchitekturen für eingebettete Systeme mitgegeben. Den Studierenden werden anhand von exemplarischen Anwendungsszenarien die typischen Entwurfsschritte vermittelt, um Software für eingebettete Systeme zu entwickeln.							
3	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Überblick und Anwendungsgebiete</li> <li>• Besonderheiten und Anforderungen</li> <li>• Grundlagen Elektronik und Hardware eingebetteter Systeme</li> <li>• Softwareentwicklungstoolchain für eingebettete Systeme</li> <li>• Mikrocontrollerprogrammierung</li> <li>• Zusammenspiel Software und Hardware</li> <li>• Entwurfsschritte</li> <li>• Spezifikation und Modellierungssprachen</li> <li>• Echtzeitbetriebssysteme</li> <li>• Realisierung und Implementierung</li> <li>• Validierung und Evaluation</li> </ul> Praktische Umsetzung eingebetteter Systeme an Anwendungsbeispielen.							
4	<b>Teilnahmevoraussetzungen</b> <b>Formal:</b> Es gilt § 17 „Fortschrittsregelung“ dieser BPO. <b>Inhaltlich:</b> Kenntnisse aus der Technischen Informatik und C++-Programmierung							
5	<b>Prüfungsgestaltung</b> Performanzprüfung							
6	<b>Voraussetzung für die Vergabe von Credit Points</b> Bestandene Modulprüfung							
7	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
8	<b>Modulbeauftragte/r</b> Prof. Dr. Dr.-Ing. Matthias König							
9	<b>Sonstige Informationen</b> Literatur wird in der Veranstaltung bekannt gegeben.							



**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Fachseminar Informatik								Kürzel FSI
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
6.1	150h	5	6. Sem.	jährlich	SoSe	1 Sem.	Pflicht	B.Sc.
<b>1</b>	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>	<b>gepl. Gruppengr.</b>	<b>Sprache</b>		
	Seminaristischer Unterricht	4 SWS / 60h	90h	Wird in LV bekannt gegeben.	35	deutsch und englisch		
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>							
	Befähigung zum selbständigen wissenschaftlichen Arbeiten mit der Erschließung von Inhalten sowie deren verständliche Darstellung von fachlichen Themen sind für den beruflichen Alltag unabdingbar. Absolventen des Moduls <ul style="list-style-type: none"> <li>• beherrschen selbstständiges Erarbeiten eines Fachthemas anhand von Fachliteratur und anderen Quellen</li> <li>• können ein Fachthema verständlich darstellen und präsentieren</li> <li>• erwerben kommunikative Kompetenz</li> <li>• vertiefen informatische Fachkompetenzen auf dem ausgewählten Themengebiet des Seminars.</li> </ul>							
<b>3</b>	<b>Inhalte</b>							
	Selbstorganisation und selbstständiges Erarbeiten eines Fachthemas <ul style="list-style-type: none"> <li>• Wissensmanagement und Literaturstudium (Recherche, Umgang mit Zitaten und Zitieren von Fachliteratur)</li> <li>• Fachliches Schreiben für die schriftliche Ausarbeitung</li> <li>• Präsentationstechnik und Rhetorik für die Präsentation des Themas</li> <li>• Diskussion im Rahmen der Seminarteilnehmer und betreuenden Dozenten</li> </ul>							
<b>4</b>	<b>Teilnahmevoraussetzungen</b>							
	keine							
<b>5</b>	<b>Prüfungsgestaltung</b>							
	Mündliche Prüfung oder Hausarbeit Oder Projektarbeit oder Exkursions- oder Tagesprotokoll oder Portfolio oder Lerntagebuch oder Parcourprüfung.							
<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b>							
	Testat Bestandene Modulprüfung							
<b>7</b>	<b>Verwendung des Moduls</b> (in folgenden Studiengängen):							
	Bachelorstudiengang Informatik							
<b>8</b>	<b>Modulbeauftragte/r</b>							
	Prof. Dr. Dominic Becking							
<b>9</b>	<b>Sonstige Informationen</b>							
	Literatur: Publikationen zum gewählten Themengebiet in deutscher und englischer Sprache							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Grundlagen der BWL								Kürzel BWL
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
4.5	150h	5	4. Sem.	jährlich	SoSe	1 Sem.	Pflicht	B.Sc.
<b>1</b>	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>	<b>gepl. Gruppengr.</b>	<b>Sprache</b>		
	Vorlesung	2 SWS / 30h	90h	Wird in LV bekannt	60	deutsch		
	Übung	2 SWS / 30h		gegeben.	30	deutsch		
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>							
	Die Studierenden erwerben Wissen um betriebswirtschaftliche Zusammenhänge im beruflichen Umfeld eines Informatikers. Sie kennen wesentliche betriebswirtschaftliche Verfahren und Grundbegriffe, haben einen Überblick zu Rechtsformen von Unternehmen, zur Investition und Finanzierung und zur Produktionsplanung und -steuerung. Sie haben einen Überblick zu Marketingstrategien. Sie haben den Bezug zwischen Algorithmen aus der Informatik und Problemen aus der Betriebswirtschaftslehre anhand ausgewählter quantitativer Beispiele (z.B. Standortplanung, Bedarfsermittlung) kennengelernt.							
<b>3</b>	<b>Inhalte</b>							
	<ul style="list-style-type: none"> <li>• Grundbegriffe der BWL</li> <li>• Unternehmensstrategien</li> <li>• Entscheidungstheorie</li> <li>• Kosten und Controlling</li> <li>• Investition und Finanzierung</li> <li>• Produktion</li> <li>• Logistik</li> <li>• Marketing</li> <li>• Personalwesen und Genderaspekte</li> </ul>							
<b>4</b>	<b>Teilnahmevoraussetzungen</b>							
	Es gilt § 17 „Fortschrittsregelung“ dieser BPO.							
<b>5</b>	<b>Prüfungsgestaltung</b>							
	Hausarbeit oder Klausur oder mündliche Prüfung							
<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b>							
	Bestandene Modulprüfung							
<b>7</b>	<b>Verwendung des Moduls (in folgenden Studiengängen):</b>							
	Bachelorstudiengang Informatik							
<b>8</b>	<b>Modulbeauftragte/r</b>							
	Prof. Dr. Dr.-Ing. Matthias König, (stellvertretend Prof. Dr.-Ing. Martin Hoffmann)							
<b>9</b>	<b>Sonstige Informationen</b>							
	Literatur wird in der Veranstaltung bekannt gegeben.							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

IT-Recht								Kürzel ITR
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
2.4	150h	5	2. Sem.	jährlich	SoSe	1 Sem.	Pflicht	B.Sc.
<b>1</b>	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>	<b>gepl. Gruppengr.</b>	<b>Sprache</b>		
	Sem. Unterricht	4 SWS / 60h	90h	Wird in LV bekannt gegeben.	35	deutsch		
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Die Studierenden kennen die wichtigsten rechtlichen Aspekte, mit denen sie während einer Tätigkeit im Gebiet der Informatik in Berührung kommen können. Insbesondere können sie einschätzen, welche Rechte und Pflichten bei Verträgen bezüglich Herstellung, Vertrieb und Nutzung von (Software/Hardware-) Produkten entstehen, mit welchen Schutzrechten geistiges Eigentum geschützt werden kann, wie Datenschutz beachtet werden muss, sowie mit welchen Folgen bei Rechtsverstößen zu rechnen ist.							
<b>3</b>	<b>Inhalte</b> Die Informationstechnologie betreffenden Inhalte sind querschnittlich im Wesentlichen aus: <ul style="list-style-type: none"> <li>• Zivilrecht und Vertragsrecht</li> <li>• Produkthaftung</li> <li>• Datenschutz</li> <li>• Strafrecht</li> <li>• Telekommunikations-, Telemedien- und Internetrecht</li> </ul> Gewerblicher Rechtsschutz (u.a. Urheber-, Patent-, Markenrecht)							
<b>4</b>	<b>Teilnahmevoraussetzungen</b> Es gilt § 17 „Fortschrittsregelung“ dieser BPO.							
<b>5</b>	<b>Prüfungsgestaltung</b> Hausarbeit oder Klausur oder mündliche Prüfung							
<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b> Teilnahme am Seminaristischen Unterricht mit Testat. Bestandene Modulprüfung							
<b>7</b>	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
<b>8</b>	<b>Modulbeauftragte/r</b> Prof. Dr. Dr.-Ing. Matthias König							
<b>9</b>	<b>Sonstige Informationen</b> Literatur wird in der Veranstaltung bekannt gegeben.							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Mathematik für Informatiker I								Kürzel MA1
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
1.1	240 h	8	1 Sem.	jährlich	WS	1 Sem.	Pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>	<b>gepl. Gruppengr.</b>	<b>Sprache</b>		
	Vorlesung	4 SWS / 60 h	90 h	Wird in LV bekannt gegeben	60	deutsch		
	Übung	2 SWS/ 30 h	60 h		20			
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Die Studierenden verstehen die mathematischen Grundbegriffe sowie die Grundlagen der Mengenlehre und Aussagenlogik. Sie können passende Beweisverfahren auswählen. Vektor- und Matrizenrechnung sowie Funktionen können von den Studierenden genutzt werden und das Lösen von linearen Gleichungssystemen kann an Beispielen angewendet werden. Die Studierenden haben die Differential- und Integralrechnung verstanden und können sie anwenden.							
3	<b>Inhalte</b> <b>Grundlagen</b> • Zahlen • Mengenlehre • Aussagenlogik • Vollständige Induktion <b>Lineare Algebra</b> • Vektoren und Vektorräume • Matrizen und lineare Abbildungen • Lineare Gleichungssysteme <b>Analysis I</b> • Folgen und Reihen • Reelle Funktionen einer Variablen • Differentialrechnung • Integralrechnung							
4	<b>Teilnahmevoraussetzungen</b> keine							
5	<b>Prüfungsgestaltung</b> Klausur mit Prüfungsvorleistung							
6	<b>Voraussetzung für die Vergabe von Credit Points</b> Bestandene Klausur							
7	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
8	<b>Modulbeauftragte/r</b> Prof. Dr. Kerstin Müller (stellvertretend Dipl. Inf. Birgit Christina George, Prof. Dr. Christoph Thiel, Dr. Jan Thies)							
9	<b>Sonstige Informationen</b> <b>Literatur</b> Hartmann, Peter: Mathematik für Informatiker, Vieweg. Manfred Brill: Mathematik für Informatiker, Hanser Verlag Bronstein, Semendjajev: Taschenbuch der Mathematik							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Mathematik für Informatiker II								Kürzel MA2
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
2.0	240 h	8	2 Sem.	jährlich	SoSe	1 Sem.	Pflicht	B.Sc.
<b>1</b>	<b>Lehrveranstaltungs- art</b>		<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>
	Vorlesung		4 SWS / 60 h	90 h	Wird in LV bekannt gegeben		60	deutsch
	Übung		2 SWS/ 30 h	60 h			20	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Die Studierenden können Funktionen mit mehreren Variablen und die zugehörige Differentialrechnung anwenden. Sie kennen lineare Differentialgleichungen und relevante Zusammenhänge aus dem Bereich der Numerik. Elementare numerische Verfahren können auf andere Situationen übertragen werden. Die Studierenden sind in der Lage geeignete Methoden zur Lösung elementarer stochastischer Probleme auszuwählen. Sie verstehen wahrscheinlichkeitstheoretische Grundbegriffe, wichtige Verteilungen und ihre Bedeutung sowie grundlegende statistische Methoden.							
<b>3</b>	<b>Inhalte</b> <b>Analysis II</b> • Lokale und globale Approximation • Differentialgleichungen • Reellwertige Funktionen mit mehreren Variablen • Differentialrechnung für Funktionen mehrerer Variablen <b>Numerik</b> • Fehler und Fehlerfortpflanzung • Elementare numerische Verfahren <b>Wahrscheinlichkeitsrechnung und Statistik</b> • Kombinatorik • Wahrscheinlichkeitsrechnung • Zufallsgrößen • Verteilungen • Statistik							
<b>4</b>	<b>Teilnahmevoraussetzungen</b> Formal: - Inhaltlich: Kenntnisse aus Modul 1.1 Mathematik 1 (MA1)							
<b>5</b>	<b>Prüfungsgestaltung</b> Klausur mit Prüfungsvorleistung							
<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b> Bestandene Klausur							
<b>7</b>	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
<b>8</b>	<b>Modulbeauftragte/r</b> Prof. Dr. Kerstin Müller (stellvertretend Dipl. Inf. Birgit Christina George, Prof. Dr. Christoph Thiel, Dr. Jan Thies)							
<b>9</b>	<b>Sonstige Informationen</b> <b>Literatur</b> Hartmann, Peter: Mathematik für Informatiker, Vieweg. Bronstein, Semendjajev: Taschenbuch der Mathematik							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Objektorientierte Programmierung								Kürzel OOP
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
1.2	210h	7	1. Sem.	jährlich	WiSe	1 Sem.	Pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>	
	Vorlesung	2 SWS / 30h	75h	Wird in LV bekannt gegeben		60	deutsch	
	Praktikum / Seminar	2 SWS / 30h	75h			15	deutsch	
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>							
	<ul style="list-style-type: none"> <li>Die Studierenden kennen Konzepte der Objektorientierung und können diese zur Entwicklung eigener Softwareanwendungen nutzen.</li> <li>Die Studierenden kennen die Programmiersprache Java und können sie zur Entwicklung eigener Software anwenden.</li> <li>Sie kennen Elemente zur Dokumentation und wenden diese in eigenen Programmen an.</li> <li>Sie können für einfache Problemstellungen objektorientierte Lösungen identifizieren und erstellen.</li> <li>Die Studierenden kennen Programmierwerkzeuge und können diese praktisch anwenden.</li> <li>Sie kennen Methoden zur Ausnahmebehandlung und wenden diese an.</li> </ul> <p>Die Studierenden lernen, Standardbibliotheken in ihren eigenen Implementierungen zielgerichtet anzuwenden.</p>							
3	<b>Inhalte</b>							
	<ul style="list-style-type: none"> <li>Einführung in objektorientiertes Design</li> <li>Einführung des Klassen- und Objektbegriffs</li> <li>Nutzung einer Entwicklungsumgebung und eines Debuggers</li> <li>Datentypen und Kontrollstrukturen</li> <li>Einführung in die Vererbung, Polymorphie</li> <li>Einführung in die Ausnahmebehandlung</li> <li>Einführung von Collection-Klassen</li> <li>Einführung in Swing</li> <li>Ansätze zum Testen</li> <li>Ansätze zum Dokumentieren (beispielsweise JavaDoc)</li> </ul>							
4	<b>Teilnahmevoraussetzungen</b>							
	keine							
5	<b>Prüfungsgestaltung</b>							
	Performanzprüfung							
6	<b>Voraussetzung für die Vergabe von Credit Points</b>							
	Bestandene Modulprüfung							
7	<b>Verwendung des Moduls</b> (in folgenden Studiengängen):							
	Bachelor Informatik							
8	<b>Modulbeauftragte/r</b>							
	Prof. Dr. Jörg Brunsmann							
9	<b>Sonstige Informationen</b>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Programmiermethoden								Kürzel PM
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
2.1	210h	7	2. Sem.	jährlich	SoSe	1 Sem.	Pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>	<b>gepl. Gruppengr.</b>	<b>Sprache</b>		
	Vorlesung	2 SWS / 30h	45h	Wird in LV bekannt gegeben	60	deutsch		
	Praktikum	3 SWS / 45h (davon 1 SWS Tutor betreut)	90h		15	deutsch		
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>							
	Nach dem Besuch der Lehrveranstaltung sind die Studierenden in der Lage, ein umfassendes Instrumentarium an Techniken und Lösungsmustern zur Softwareentwicklung in der Programmiersprache Java einzusetzen. Der erfolgreiche Studierende kann wichtige Standardarchitekturmuster passend zur Aufgabe auswählen und einsetzen. Die Studierenden sind in der Lage, komplexere Anwendungen und eigene Bibliotheken zu entwickeln. Die Studierenden beherrschen grundlegende Techniken und Workflows zur Quellcode-Versionsverwaltung und sind in der Lage, diese in Projekten aktiv anzuwenden. Die Studierenden erkennen „Bad Smells“ und sind in der Lage, diese mit Hilfe von Refactoring unter Absicherung durch selbst formulierte Unit-Tests zu beheben.							
3	<b>Inhalte</b>							
	<ul style="list-style-type: none"> <li>• Modellierung, Pattern, OOD: <ul style="list-style-type: none"> <li>– Vertiefung objektorientierter Entwurf/Design</li> <li>– Vertiefung Klassendiagramme mit UML</li> <li>– Basis-Entwurfsmuster (Strategy, Decorator, Template, Factory-Method, Observer)</li> <li>– Vererbung vs. Komposition vs. Generics</li> </ul> </li> <li>• Programmiermethoden: <ul style="list-style-type: none"> <li>– Versionsverwaltung am Beispiel Git, Branching-Strategien und Workflows</li> <li>– Systematischer Unit-Test mit JUnit</li> <li>– Systematische Dokumentation mit JavaDoc</li> <li>– Coding Conventions, Bad Smells, CheckStyle</li> <li>– Refactoring mit Eclipse</li> <li>– Buildsysteme (ANT, Maven, Gradle)</li> <li>– Einführung in die Nutzung von CI</li> </ul> </li> <li>• Fortgeschrittene Java-Kenntnisse: <ul style="list-style-type: none"> <li>– Annotationen und Enumerationen</li> <li>– Reguläre Ausdrücke mit Java</li> <li>– Generische Programmierung (Generics)</li> <li>– Logging unter Java</li> <li>– Konfiguration (Properties, Preferences, CLI)</li> <li>– Anonyme und innere Klassen</li> <li>– Reflection, Serialisierung</li> <li>– Einführung in parallele Programmierung mit Java: Threads, synchronize, wait, notify</li> <li>– Java8: Defaultmethoden, Methodenreferenzen, Funktions-Interfaces, funktionale Sprachelemente</li> <li>– Ereignisorientierte Programmierung (Swing oder JavaFX, Java2D), Modelle und Event-Hierarchien</li> <li>– Java und XML: JAXB</li> <li>– Nutzung von Softwarebibliotheken (APIs): beispielsweise Apache CLI, Apache POI, JFreeChart</li> </ul> </li> </ul>							
4	<b>Teilnahmevoraussetzungen</b>							
	keine							
5	<b>Prüfungsgestaltung</b>							
	Performanzprüfung oder Klausur oder Hausarbeit oder Parcoursprüfung (Stationenprüfung)							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b> Testat und bestandene Modulprüfung
<b>7</b>	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik
<b>8</b>	<b>Modulbeauftragte/r</b> Prof. Dr.-Ing. Carsten Gips
<b>9</b>	<b>Sonstige Informationen</b> <ul style="list-style-type: none"> <li>• Deitel, Deitel: "Java – How to Program", Pearson Education Limited, 2012</li> <li>• Bloch, J.: "Effective Java: A Programming Language Guide", Addison-Wesley, 2011</li> <li>• Urma, Fusco, Mycroft: "Java 8 in Action", Manning Publications, 2014</li> <li>• Chacon, Straub: "Pro Git", Apress, 2014</li> <li>• Robert Martin: "Clean Code", Prentice Hall, 2008</li> <li>• Martin Fowler et al.: "Refactoring", Addison Wesley, 1999</li> <li>• Roy Oshero: "The Art of Unit Testing", Manning, 2013</li> <li>• Kent Beck: "Test Driven Development", Addison-Wesley, 2002</li> <li>• Gamma et al.: "Design Patterns", Addison-Wesley, 2011</li> <li>• Ulllenboom, C.: "Java ist auch eine Insel", Rheinwerk-Verlag, 2016</li> </ul>



**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Software Engineering								Kürzel SE
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
3.0	210h	7	3. Sem.	jährlich	WiSe	1 Sem.	Pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>	
	Vorlesung	2 SWS / 30h	75h	Wird in LV bekannt gegeben		60	deutsch	
	Praktikum / Seminar	2 SWS / 30h	75h			15	deutsch	
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>							
	<ul style="list-style-type: none"> <li>Die Studierenden erlernen das systematische und strukturierte Vorgehen zur erfolgreichen Planung und Durchführung eines Softwareentwicklungsprojekts und wenden dazu agile Vorgehensmodelle praktisch an.</li> <li>Sie lernen relevante Phasen von der Anforderungsanalyse über Grobdesign, Feindesign, Implementierung bis hin zur Qualitätssicherung kennen.</li> <li>Sie lernen im Detail die Notationselemente und Diagrammtypen des UML-Standards kennen und wenden diese für ein Softwareprojekt an.</li> <li>Die Studierenden lernen Architektur- und Entwurfsmuster sowie JUnit-Tests kennen</li> </ul>							
3	<b>Inhalte</b>							
	<ul style="list-style-type: none"> <li>Einführung in Software Engineering</li> <li>UML-Diagramme (z.B. Modellierung von Unternehmensprozesse mit Aktivitätsdiagrammen)</li> <li>Vorgehensmodelle (Wasserfallmodell, agile Vorgehensmodelle wie Scrum und Extreme Programming)</li> <li>Anforderungsanalyse (Stakeholder, Ziele, Use Cases, Ableitung funktionaler Anforderungen, nicht funktionale Anforderungen, Lasten- und Pflichtenheft)</li> <li>Grobdesign (Systemarchitektur, Ableitung der grundlegenden Klassen, Methoden, Sequenzdiagramm, Überlegungen zur Oberflächenentwicklung)</li> <li>Programmgenerierung (Übersetzung von Klassen und Assoziationen, Arten der Objektzugehörigkeit, Software-Architektur)</li> <li>Feindesign (Details im Kleinen, Model View Controller, GoF-Pattern)</li> <li>Implementierungen (Verteilte Systeme, Bibliotheken, Komponenten, Frameworks, Persistente Datenhaltung)</li> <li>SW-Qualitätssicherung (Zusicherungen, Unit-Tests, Testverfahren, Metriken)</li> </ul>							
4	<b>Teilnahmevoraussetzungen</b>							
	Formal: -, Inhaltlich: Kenntnisse aus Objektorientierte Programmierung (OOP), Programmiermethoden (PM)							
5	<b>Prüfungsgestaltung</b>							
	Performanzprüfung							
6	<b>Voraussetzung für die Vergabe von Credit Points</b>							
	Bestandene Modulprüfung							
7	<b>Verwendung des Moduls</b> (in folgenden Studiengängen):							
	Bachelor Informatik							
8	<b>Modulbeauftragte/r</b>							
	Prof. Dr. Jörg Brunsmann							
9	<b>Sonstige Informationen</b>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Softwareprojekt								Kürzel SWP
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
4.0	150h	5	4. Sem.	jährlich	SoSe	1 Sem.	Pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>	<b>gepl. Gruppengr.</b>	<b>Sprache</b>		
	Praktikum / Seminar	4 SWS / 60h	90h	Wird in LV bekannt gegeben	15	deutsch		
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> In enger Verzahnung mit dem Modul "Projektmanagement" Die Studierenden wenden die wesentlichen Grundlagen des Projektmanagements an. Sie planen ein größeres Softwareprojekt, führen es durch, leiten es und dokumentieren und präsentieren regelmäßig Projektfortschritt und –ergebnisse. Sie entscheiden sich begründet in dem gegebenen Projektzusammenhang für ein Vorgehensmodell. Sie wenden Vorgehensweise, Organisationsformen und Methoden eines anerkannten formalen Projektmanagementsystems auf ihr Projekt an. Sie realisieren ein Projekt in einer größeren Projektgruppe (ca. 8 Personen) mit Rollenverteilung.							
3	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Einrichtung und Durchführung eines Projektes</li> <li>• Erstellung eines Pflichtenhefts auf Basis der Vorgaben eines fiktiven „Kunden“, Durchführung einer Aufwandsschätzung auf Basis dieses Pflichtenhefts</li> <li>• Aufstellen eines Projektplanes und Verfahren zur Projektverfolgung und des Risikomanagements</li> <li>• Implementierung im Team (Version-Management, Build-Verfahren, Abstimmungsprozesse, Schnittstellen)</li> <li>• Präsentation von Ergebnissen und Zwischenergebnissen</li> <li>• Einsatz aktueller Technologien zur Implementierung der Anwendung</li> <li>• Abschlusspräsentation</li> </ul>							
4	<b>Teilnahmevoraussetzungen</b> <b>Formal:</b> - <b>Inhaltlich:</b> Kenntnisse aus Objektorientierte Programmierung (OOP) Es gilt § 17 „Fortschrittsregelung“ dieser BPO.							
5	<b>Prüfungsgestaltung</b> Projektarbeit							
6	<b>Voraussetzung für die Vergabe von Credit Points</b> Bestandene Modulprüfung							
7	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
8	<b>Modulbeauftragte/r</b> Dozenten im Studiengang Informatik (Becking, Behrens, Brunsmann, George, Gips, Hoffmann, Kreienkamp, König, Müller, Thiel)							
9	<b>Sonstige Informationen</b>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Software-Projektmanagement								Kürzel SPM
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
3.4	150h	5	3. Sem	jährlich	WiSe	1 Sem	Pflicht	B.Sc.
<b>1</b>	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>	<b>gepl. Gruppengr.</b>	<b>Sprache</b>		
	Seminaristischer Unterricht	4 SWS / 60h	90h	Wird in LV bekannt gegeben	35	deutsch		
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Die Studierenden wenden die wesentlichen Grundlagen des Projektmanagements sicher auf beispielhafte Projekte an. Sie sind in der Lage, Projekte zu planen, durchzuführen, in Teilen zu leiten sowie Projektfortschritt und -ergebnisse zu dokumentieren und zu präsentieren. Sie können sich begründet in einem Projektzusammenhang für ein Vorgehensmodell entscheiden. Sie kennen Vorgehensweise, Organisationsformen und Methoden eines anerkannten formalen Projektmanagementsystems.							
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Projekte als Problemlösungsprozesse</li> <li>• Gründung, Organisation und Strukturierung von Projekten</li> <li>• Projektplanung</li> <li>• Projektsteuerung</li> <li>• Management von Software-Projekten</li> <li>• Werkzeuge im Projektmanagement</li> <li>• Kommunikation und Dokumentation als Querschnittsaufgabe</li> <li>• Qualitätssicherung</li> <li>• Projektmanagementsysteme</li> </ul>							
<b>4</b>	<b>Teilnahmevoraussetzungen</b> keine							
<b>5</b>	<b>Prüfungsgestaltung</b> Mündliche Prüfung oder Hausarbeit oder Projektarbeit oder Praktikums-, Exkursions- oder Tagesprotokoll oder Portfolio oder Lerntagebuch oder Parcourprüfung							
<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b> Testat Bestandene Modulprüfung							
<b>7</b>	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
<b>8</b>	<b>Modulbeauftragte/r</b> Prof. Dr. Dominic Becking							
<b>9</b>	<b>Sonstige Informationen</b> Literatur: <ul style="list-style-type: none"> <li>• o.V., Managing Successful Projects with PRINCE2® 2009 Edition, Axelos 2009</li> <li>• o.V., A Guide to the Project Management Body of Knowledge (PMBOK® Guide), PMI 2012</li> </ul>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Systemprogrammierung								Kürzel SP
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
3.3	240h	8	3. Sem.	jährlich	WiSe	1 Sem.	Pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>	<b>gepl. Gruppengr.</b>	<b>Sprache</b>		
	Vorlesung	3 SWS / 45h	50h	Wird in LV bekannt gegeben	60	deutsch		
	Praktikum	3 SWS / 45h	100h		15	deutsch		
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>							
	Die Studierenden erwerben umfassende Kompetenzen zur Entwicklung systemnaher Software in den aktuellen Systemprogrammiersprachen C und C++ unter Berücksichtigung gebräuchlicher Standards (z.B. ANSI-C/C11 und C++14/C++17). Sie beherrschen aktuelle Tools in diesem Umfeld, beispielsweise die Gnu-Compiler (gcc, g++) samt Debugger (gdb) und Make, und kennen verschiedene aktuelle Standardbibliotheken. Die Programmentwicklung unter Nutzung wesentlicher Teile der UNIX/Linux-Programmierschnittstelle (POSIX) wird beherrscht. Die Studierenden können dieses Wissen selbständig auf komplexere Aufgabenstellungen anwenden (Praktikumsaufgaben).							
3	<b>Inhalte</b>							
	<ul style="list-style-type: none"> <li>• Grundlagen C/C++ <ul style="list-style-type: none"> <li>– Typen, Ausdrücke, Operatoren, Kontrollfluss</li> <li>– Strukturen und Enumerationen, typedef</li> <li>– Bitoperationen</li> <li>– Funktionen, Deklaration vs. Definition, Prototypen, Call-by-Value</li> <li>– Sichtbarkeiten und Scopes, globale vs. lokale Variablen, static vs. extern, dateiübergreifend</li> <li>– Speicherklassen</li> <li>– Referenzen in C++</li> <li>– Überladen von Operatoren und Funktionen in C++</li> <li>– Nutzung eines Debuggers, z.B. gdb</li> <li>– Unit-Test in C++, z.B. mit cppunit oder googletest</li> </ul> </li> <li>• Objektorientierte Programmierung in C++ <ul style="list-style-type: none"> <li>– Klassen, Konstruktoren, Destruktoren, Copy-Konstruktor, Zuweisungsoperator</li> <li>– Friends</li> <li>– Operatoren</li> <li>– Trennung Interface und Implementierung</li> <li>– Vererbung, Polymorphie, virtual, Slicing, abstrakte Klassen, Mehrfachvererbung</li> </ul> </li> <li>• Modulare Programmierung <ul style="list-style-type: none"> <li>– Aufteilung auf Header- und Implementierungsdateien</li> <li>– One Definition Rule</li> <li>– Präprozessor: Include, Makros, bedingte Übersetzung, Konstanten</li> <li>– Statische und dynamische Bibliotheken, Linker</li> <li>– Makefiles</li> </ul> </li> <li>• Speicherverwaltung <ul style="list-style-type: none"> <li>– Speicherverwaltung unter Linux, virtueller Speicher, Stack vs. Heap</li> <li>– Pointer und Adressen, Deklaration, Dereferenzierung, Zuweisung</li> <li>– Dynamische Speicherverwaltung mit malloc/free und new/delete</li> <li>– Probleme bei Speicherverwaltung: Memory Leaks, Stale Pointer, Double Delete</li> <li>– Call-by-Reference in C durch Pointer</li> <li>– Zusammenhang Pointer und Arrays, mehrdimensionale Arrays, CMD-Parameter</li> <li>– Adress-Arithmetik</li> <li>– C-Strings und Funktionen aus der C-Std-Lib (z.B. strcpy, strcat, strtok)</li> <li>– Funktionspointer</li> <li>– SmartPointer in C++</li> <li>– Lesen komplexer Deklarationen</li> </ul> </li> <li>• Ein- und Ausgabe, Umgang mit Verzeichnissen <ul style="list-style-type: none"> <li>– Systemfunktionen unter Linux, Datei-Abstraktion, Standard-E/A-Kanäle</li> <li>– Umgang mit Dateien unter C, scanf/printf</li> </ul> </li> </ul>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

	<ul style="list-style-type: none"> <li>– Streams in C++, Fehlerzustände, Manipulation</li> <li>• Fehlerbehandlung <ul style="list-style-type: none"> <li>– Signalisierung von Fehlern in der Linux-Systemschnittstelle (Rückgabewert, errno), abort/exit/atexit</li> <li>– Umgang mit Exceptions (C++)</li> <li>– Assertions</li> </ul> </li> <li>• Standards: ANSI C vs. C11, C++11 vs. C++14 vs. C++17</li> <li>• Metaprogrammierung mit Templates (Funktionen, Klassen)</li> <li>• Einführung in Standardbibliotheken (z.B. STL, Boost)</li> <li>• C++14: Move-Semantik, SmartPointer</li> <li>• Prozess- und Thread-Manipulation (Erzeugen, Beenden, Zustände, Zombies/Waise)</li> <li>• Interprozess-Kommunikation: Signale und Sockets, Überblick weitere IPC-Formen</li> <li>• Zeit (Kalender, Uhrzeit, Zeitmessung und Timer)</li> <li>• Einbettung/Anbindung anderer Sprachen (z.B. Python, Lua)</li> <li>• Nutzung von Bibliotheken, z.B. SQLite3, libXML2, libCurl</li> <li>• Systemnahe Programmierung unter Linux auf dem Raspberry Pi (C/C++)</li> <li>• Dokumentation mit Doxygen</li> <li>• Sichere und defensive Programmierung</li> <li>• Wechselnde Inhalte der Praktika zu aktuellen Themen</li> </ul>
<b>4</b>	<b>Teilnahmevoraussetzungen</b> keine
<b>5</b>	<b>Prüfungsgestaltung</b> Performanzprüfung oder Klausur oder Hausarbeit oder Parcoursprüfung (Stationenprüfung)
<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b> Testat und bestandene Modulprüfung
<b>7</b>	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik
<b>8</b>	<b>Modulbeauftragte/r</b> Prof. Dr.-Ing. Carsten Gips
<b>9</b>	<b>Sonstige Informationen</b> <ul style="list-style-type: none"> <li>• Breyman, U.: "Der C++ Programmierer", Hanser, 2011.</li> <li>• Scott Meyers: "Effective Modern C++", O'Reilly, 2014</li> <li>• Klemens, B.: "21st Century C", O'Reilly, 2014</li> <li>• Brian Kernighan, Dennis Ritchie: "The C Programming Language", Prentice Hall, 2000</li> <li>• Love, O.: "Linux System Programming", O'Reilly Media, 2013</li> <li>• Kerrisk, M.: "The Linux Programming Interface", no starch press, 2011</li> </ul>

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Technical English								Kürzel TE
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
5.0	150h	5	5. Sem.	jährlich	WiSe	1 Sem.	Pflicht	B.Sc.
<b>1</b>	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>	<b>gepl. Gruppengr.</b>	<b>Sprache</b>		
	Sem. Unterricht	4 SWS / 60h	90h	Wird in LV bekannt gegeben	35	englisch		
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Die Teilnehmer sollen in der Lage sein sich mit technischem Englisch auszudrücken. Sie sollen diese erworbenen Sprachkenntnisse in einem internationalen Umfeld sowohl schriftlich als auch gesprochen anwenden können. Die Studierenden erlernen hierzu die spezifische Terminologie, welche im Bereich Informatik angewendet wird, als auch allgemeines technisches Englisch und werden in der Lage sein, Prozesse zu beschreiben und zu erklären, technische Probleme zu diskutieren und IT-Lösungen zu vermitteln.							
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Computer Hardware: Ein- und Ausgabemedien, Drucker, Speichermedien</li> <li>• Wesentliche und kreative Software: Betriebssystem, Textverarbeitung, Tabellenkalkulation und Datenbanken, Desktop-Publishing, Multimedia</li> <li>• Programmierung: Programmiersprachen, Java, Webgestaltung</li> <li>• Zukunftstechnologien: Kommunikationssysteme, Netzwerke, Videospiele</li> <li>• Internet: E-Mail, das Web, Videokonferenzführung</li> </ul> Beschäftigungsmöglichkeiten im Bereich IKT							
<b>4</b>	<b>Teilnahmevoraussetzungen</b> Es gilt § 17 „Fortschrittsregelung“ dieser BPO.							
<b>5</b>	<b>Prüfungsgestaltung</b> Klausur							
<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b> Teilnahme am Seminaristischen Unterricht mit Testat Bestandene Klausur							
<b>7</b>	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
<b>8</b>	<b>Modulbeauftragte/r</b> Prof. Dr.-Ing. Martin Hoffmann							
<b>9</b>	<b>Sonstige Informationen</b>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Technische Informatik								Kürzel TI
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
1.4	150 h	5	1. Sem.	jährlich	WiSe	1 Sem.	Pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>		<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppeng r.</b>	<b>Sprache</b>
	Vorlesung		2 SWS / 30h	90h	Wird in LV bekannt gegeben		60 (V)	deutsch
	Übung		2 SWS / 30h				30 (Ü)	deutsch
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Die Studierenden können nach Abschluss der Veranstaltung <ul style="list-style-type: none"> <li>• in der Elektrotechnik einfache Gleichstromkreise mit Hilfe des Ohmschen Gesetzes und der Kirchhoffschen Regeln berechnen;</li> <li>• Eigenschaften und Einsatzgebiete verschiedener Prozessortypen zuordnen;</li> <li>• den grundlegenden Aufbau und die funktionalen Einheiten eines Prozessors darstellen und deren Funktionen beschreiben;</li> <li>• grundlegende Prozessorarchitekturen benennen, deren Eigenschaften bzw. Vor- und Nachteile anführen, vorliegende Architekturen (im Blockschaltbild) identifizieren;</li> <li>• die wichtigsten Befehls- und Adressierungsarten benennen, Befehle anhand des Datenblattes verstehen und in Assembler/Maschinencode umsetzen;</li> <li>• die Prinzipien der Fehlererkennung und Fehlerkorrektur darstellen, Aufbau und Eigenschaften eines Hamming-Codes herleiten und beispielhaft einsetzen;</li> <li>• Speichertechnologien gegenüberstellen und deren Einsatzbeispiele benennen;</li> <li>• Boole'schen Rechenregeln anwenden;</li> <li>• digitale Schaltpläne aus Funktionsgleichungen erstellen und umgekehrt; Funktionsgleichungen</li> <li>• in Normalform aus Wertetabellen lesen und mittels KV-Diagramm minimieren;</li> <li>• den prinzipiellen Aufbau bistabiler Kippstufen beschreiben, deren Klassifizierungen voneinander abgrenzen; Eigenschaften und Einsatzbeispiele von RS-, D-, JK- und T-Flipflop benennen und deren Arbeitsweise anhand von Impulsdigrammen erläutern.</li> </ul>							
3	<b>Inhalte</b> Grundlagen des elektrischen Gleichstromkreises <ul style="list-style-type: none"> <li>• Ohmsches Gesetz</li> <li>• Kirchhoffsche Sätze</li> </ul> Grundlagen der Rechnerarchitekturen <ul style="list-style-type: none"> <li>• Aufbau von Prozessoren mit Prozessortypen, Architekturen, Steuer-/Rechenwerk und Register,</li> <li>• Einführung in die Hardware-nahe Programmierung</li> <li>• Programm- und Datenspeicher mit Speicherorganisation, Speichertechnologien</li> <li>• Fehlererkennung und -korrektur in der Datenübertragung</li> </ul> Grundlagen der Digitaltechnik <ul style="list-style-type: none"> <li>• Boolesche Algebra, Normalform und Minimierung</li> <li>• Schaltnetze und Schaltwerke</li> </ul>							
4	<b>Teilnahmevoraussetzungen</b> keine							
5	<b>Prüfungsgestaltung</b> Klausur							
6	<b>Voraussetzung für die Vergabe von Credit Points</b> Bestandene Klausur							
7	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
8	<b>Modulbeauftragte/r</b> Angela Kreienkamp (MPrComp)							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

<b>9</b>	<b>Sonstige Informationen</b> Literatur (z.B.): <ul style="list-style-type: none"><li>• Elektrotechnik Grundlagen, Steffen, Bausch</li><li>• Digitaltechnik - Ein Lehr- und Übungsbuch, Weitowitz, Urbanski</li><li>• Mikrocontroller und Mikroprozessoren, Brinkschulte, Ungerer</li><li>• Logischer Entwurf digitaler Systeme, Liebig</li><li>• Mathematik sehen und verstehen, Haftendorn</li></ul>
----------	---



**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Theoretische Informatik								Kürzel THI
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
2.3	150h	5	2. Sem.	jährlich	SoSe	1 Sem.	Pflicht	B.Sc.
<b>1</b>	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>	<b>gepl. Gruppengr.</b>	<b>Sprache</b>		
	Vorlesung	2 SWS / 30h	45h	Wird in LV bekannt gegeben	60	deutsch		
	Übung	2 SWS / 30h	45h		30	deutsch		
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Die Studierenden können <ul style="list-style-type: none"> <li>• Automaten und Grammatiken entwickeln</li> <li>• zu gegebenen Automaten, Kellerautomaten, unterschiedlichen Arten von Grammatiken und Turingmaschinen die entsprechende Sprache angeben und umgekehrt</li> <li>• die Chomsky-Hierarchie aufstellen und den Klassen Sprachen und Automaten zuordnen</li> <li>• Probleme der Berechenbarkeit, der Entscheidbarkeit und das Halteproblem erklären und diskutieren</li> <li>• das P-NP-Problem an Hand von Beispielen diskutieren</li> </ul>							
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Automaten (mit Ausgabe, deterministisch, nicht deterministisch)</li> <li>• reguläre Ausdrücke</li> <li>• Grammatiken, kontextfreie Sprachen</li> <li>• Kellerautomaten</li> <li>• kontextsensitive und Typ 0 - Sprachen, Turingmaschine</li> <li>• Chomsky-Hierarchie</li> <li>• Berechenbarkeit, Entscheidbarkeit, Church'sche These, Halteproblem,</li> <li>• Komplexitätstheorie, P-NP-Problem</li> </ul>							
<b>4</b>	<b>Teilnahmevoraussetzungen</b> keine							
<b>5</b>	<b>Prüfungsgestaltung</b> Klausur							
<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b> Bestandene Klausur							
<b>7</b>	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
<b>8</b>	<b>Modulbeauftragte/r</b> Dipl.-Inf. BC George							
<b>9</b>	<b>Sonstige Informationen</b>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Usability und Datenvisualisierung								Kürzel UDV
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q- Niveau
6.0	150 h	5	6. Sem.	jährlich	SoSe	1 Sem.	Pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>	<b>gepl. Gruppengr.</b>	<b>Sprache</b>		
	Sem. Unterricht	4 SWS / 60h	90h	Wird in LV bekannt gegeben	35	deutsch		
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Die Studierenden können nach Abschluss der Veranstaltung <ul style="list-style-type: none"> <li>• die Möglichkeiten der Wahrnehmung und das Verarbeiten von Informationen beim Menschen beschreiben;</li> <li>• gängige Techniken zur Eingabe und Darstellung von Informationen benennen und beschreiben;</li> <li>• diverse Interaktionsmodelle darstellen und am Beispiel anwenden;</li> <li>• unterschiedliche Prinzipien und Techniken der Datenvisualisierung erläutern;</li> <li>• einzelne Usability Konzepte und Prinzipien beschreiben und vergleichen;</li> <li>• verschiedenartige Verfahren zur Usability Effizienzmessung beschreiben, vergleichen und anwenden;</li> <li>• wesentliche Methoden des Design Prozesses beschreiben und vergleichen;</li> <li>• maßgebende Methoden der Evaluation benennen, deren Eigenschaften erläutern und am Beispiel anwenden</li> </ul>							
3	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Psychologische, physikalische und theoretische Aspekte der Interaktion des Menschen in seiner Umgebung (HCI-Human Computer Interaction)</li> <li>• Gestaltungsprinzipien und Techniken zur Visualisierung von Daten</li> <li>• Modelle und Methoden des Usability Engineerings</li> <li>• Methoden zur Evaluation von Usability</li> </ul>							
4	<b>Teilnahmevoraussetzungen</b> keine							
5	<b>Prüfungsgestaltung</b> Klausur und/oder Hausarbeit							
6	<b>Voraussetzung für die Vergabe von Credit Points</b> Teilnahme am Seminaristischen Unterricht mit Testat Bestandene Klausur/Hausarbeit							
7	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
8	<b>Modulbeauftragte/r</b> Prof. Dr. Kerstin Müller							
9	<b>Sonstige Informationen</b> Literatur (z.B.): <ul style="list-style-type: none"> <li>• The Laws of Simplicity, J. Maeda</li> <li>• Human-Computer Interaction, A. Dix, J. Finlay, G. Abowd, R. Beale</li> <li>• Human-Centered Visualization Environments, A. Kerren, A. Ebert, J. Meyer</li> <li>• Interaction Design, H. Sharp, Y. Rogers, J. Preece</li> </ul>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Webbasierte Anwendungen								Kürzel WBA
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
4.4	150h	5	4. Sem.	jährlich	SoSe	1 Sem.	Pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>	
	Vorlesung	2 SWS / 30h	45h	Wird in LV bekannt gegeben		60	deutsch	
	Praktikum	2 SWS / 30h	45h			15	deutsch	
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Webbasierte Systeme sind eine der verbreitetsten Ausprägungen verteilter Informationssysteme und sind heute weder aus dem privaten noch aus dem beruflichen Bereich wegzudenken. Die Studierenden lernen diese professionell zu planen, zu realisieren und zu beurteilen. Sie lernen die vom W3C standardisierten Basistechnologien kennen und erwerben die Fähigkeit, diese problemadäquat einzusetzen. Sie bekommen einen Überblick über aktuelle open source Frameworks für die professionelle Webentwicklung und nutzen eine Auswahl davon zur Konzipierung und Implementierung eigener Anwendungssysteme.							
3	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Einführung, Klassifikation von Web-Anwendungen, Architekturen</li> <li>• Grundlagen (HTTP, Sessionmanagement, Standardisierung, W3C)</li> <li>• Webseiten gestalten (Cascading Stylesheets, HTML5)</li> <li>• Clientseitige Technologien: JavaScript, Ajax, DOM, aktuelle Bibliotheken und Frameworks</li> <li>• Serverseitige Mehrschichtenarchitekturen, Frameworks zu deren Umsetzung: z.B. JSF,</li> <li>• Applikationsserver (Aufgaben, Dienste, Beispiele): z.B. Glassfish</li> <li>• Webservices z.B. REST</li> <li>• Datenaustauschformate z.B. JSON</li> </ul>							
4	<b>Teilnahmevoraussetzungen</b> <b>Formal:</b> - <b>Inhaltlich:</b> Kenntnisse in Programmieren in Java, Softwaretechnik, Einführung in die Programmierung mit Skriptsprachen, Datenbanken							
5	<b>Prüfungsgestaltung</b> Performanzprüfung							
6	<b>Voraussetzung für die Vergabe von Credit Points</b> Bestandene Modulprüfung							
7	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
8	<b>Modulbeauftragte/r</b> Prof. Dr.-Ing. Grit Behrens							
9	<b>Sonstige Informationen</b> Literatur: Kurz, Marinschek: „JavaSever Faces 2.2: Grundlagen und erweiterte Konzepte“, dpunkt 2013 Schießer, Schmollinger „Workshop in JavaEE: Ein praktischer Einstieg in die Java Enterprise Edition mit dem Web Profile“, dpunkt 2014 Dean Cemron „HTML5, JavaScript und jQuery“, dpunkt 2015 Somin Timms „Mastering JavaScript Design Patterns“, Packt Publishing 2016							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodul aus Liste 1 „Methoden der Informatik“ Datenbanken II: Architekturen und Implementierungstechniken								Kürzel DB2
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
5.10	150h	5	5./6. Sem.	halbjährlich bei Nachfrage	SoSe/WS bei Nachfrage	1 Sem.	Wahl- pflicht	B.Sc.
<b>1</b>	<b>Lehrveranstaltungs- art</b>		<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>
	Vorlesung		2 SWS / 30h	45h	Wird in LV bekannt gegeben		60	deutsch
	Praktikum		2 SWS / 30h	45h			15	deutsch
<b>2</b>	<b>Lernergebnisse (learning outcomes)/ Kompetenzen</b> Datenbanksysteme sind hochkomplexe Softwaresysteme, zu deren sicherer Anwendung und Nutzung ein vertieftes Wissen der Implementierung und Administration gehört. Die Studierenden können Probleme in Datenbanksystemen mit Hilfe ihrer erworbenen theoretischen Kenntnisse in fortgeschrittenen Datenbanktechniken sicher analysieren und erklären und Lösungsansätze aus der Theorie ableiten. Sie fällen begründete Entscheidungen für die Anwendung dieser Techniken. Sie können Datenbanken installieren und administrieren und dabei die Anforderungen der Anwendungssoftware analysieren und umsetzen. Sie können die Performanz von Datenbanken analysieren und durch geeignete technische Maßnahmen steigern. Sie können fortgeschrittene SQL formulieren und sinnvoll anwenden. Sie können SQL zur Performanzsteigerung analysieren und optimieren (SQL-Tuning).							
<b>3</b>	<b>Inhalte</b> Folgende Themenbereichen stehen exemplarisch für mögliche Inhalte: <ul style="list-style-type: none"> <li>• Architektur von Datenbanksystemen</li> <li>• Verwaltung des Hintergrundspeichers</li> <li>• Pufferverwaltung</li> <li>• Dateiorganisation und Zugriffsstrukturen</li> <li>• Spezielle Indexstrukturen</li> <li>• Basisalgorithmen für Datenbankoperationen</li> <li>• Fortgeschrittene SQL</li> <li>• Optimierung von Anfragen</li> <li>• Transaktionsmodelle</li> <li>• Transaktionsverwaltung</li> <li>• Wiederherstellung und Datensicherung</li> <li>• Moderne Datenbankparadigma</li> </ul> Aktuelle Probleme der Datenbanktechnik							
<b>4</b>	<b>Teilnahmevoraussetzungen</b> Formal: - Inhaltlich: Inhalte des Moduls 3.2 „Datenbanken I“ (DB1)							
<b>5</b>	<b>Prüfungsgestaltung</b> Mündliche Prüfung oder Hausarbeit oder Projektarbeit oder Praktikums-, Exkursions- oder Tagesprotokoll oder Portfolio oder Lerntagebuch oder Parcourprüfung							
<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b> Praktikum mit Testat Bestandene Modulprüfung							
<b>7</b>	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
<b>8</b>	<b>Modulbeauftragte/r</b> Prof. Dr. Dominic Becking							
<b>9</b>	<b>Sonstige Informationen</b> Literatur: <ul style="list-style-type: none"> <li>• Saake, G., Sattler, K.-U., Datenbanken: Implementierungstechniken, Heidelberg 2011</li> <li>• Härder, Th., Rahm, E., Datenbanksysteme: Konzepte und Techniken der Implementierung, Berlin 2001</li> <li>• Aktuelle Literatur zu neueren Datenbanktechnologien</li> </ul>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodul aus Liste 1 „Methoden der Informatik“ Einführung in Audiovisual Computing								Kürzel AV1
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
5.11	150h	5	5./6. Sem.	halbjährlich bei Nachfrage	SoSe/WS bei Nachfrage	1 Sem.	Wahl- pflicht	B.Sc.
1a	<b>Lehrveranstaltungs- art</b>		<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>
	Vorlesung		2 SWS / 30h	45h	Wird in LV bekannt gegeben		60	deutsch
Praktikum		2 SWS / 30h	45h			15	deutsch	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>							
<p>Als Teilgebiet der Informatik ist die Medieninformatik stark interdisziplinär ausgerichtet. Hintergrund für ihre Entstehung ist die seit Anfang der 1990er-Jahre zunehmende Digitalisierung von Text, Bild und Video. Rund um den Multimedia-Begriff entstanden unzählige neue Technologien und Anwendungen sowie die dazu entsprechenden Märkte, Tätigkeitsfelder und Berufsbilder. Die Erzeugung, Bearbeitung, Speicherung und Verbreitung von audiovisuellen Signalen ist einer der zentralen Aspekte der Medieninformatik. Das Spezialgebiet Audiovisual Computing beschäftigt sich mit den Wechselwirkungen der technischen Grundlagen und Möglichkeiten einerseits und der künstlerischen Gestaltung andererseits. Die Musikinformatik als ein Teilgebiet des Audiovisual Computing z.B. befasst sich mit allen computerbasierten Techniken und der Entwicklung von Anwendungen zur Komposition, Produktion, Vertrieb, Abrechnung/Lizenzen und dem Genuss von Musik und anderen Audioprodukten. Darüber hinaus sind spezielle Aspekte des Musikmanagements, der Musikwirtschaft und der technischen Unterstützung kreativer Prozesse Musikschaffender Gegenstand des Fachgebiets.</p> <p>Die Studierenden erarbeiten sich anhand eines komplexen Projektes aus dem Bereich Audiovisual Computing einen wissenschaftlichen Zugang zu diesem wichtigen Teilgebiet der Medieninformatik.</p> <p>Die Studierenden erleben und beschreiben Musik, Video und Bild als komplexes kulturelles und technisches Phänomen. Sie analysieren Aspekte der Generierung, Produktion und Vertrieb von Audiovisuellen Medien in Bezug auf die Rolle der IT. Die Studierenden beziehen Erkenntnisse über die Musik und visueller Kommunikation als universelles kulturelles Phänomen in Ihre Überlegungen ein und machen sich dazu mit wissenschaftlicher Literatur aus Anthropologie, Psychologie und den Kulturwissenschaften vertraut. Sie nutzen Standardprogramme der Audiovisual Computing und produzieren eigene Musik und audiovisuelle Kunstwerke.</p> <p>In Absprache mit dem Dozenten wählen die Studierenden Projektthemen aus und bearbeiten diese über ein Semester als Projektgruppe. Sie recherchieren den Stand der Technik und Wissenschaft, formulieren ein Entwicklungsziel und erarbeiten das erforderliche Skillset. Sie bedienen sich aktueller Projektmanagementmethoden und –tools. Sie implementieren ausgewählte Teile der Modellierung in funktionierende Software. Sie präsentieren Ergebnisse sowohl in akademischen als auch musikalischen und visuell-künstlerischen Formaten.</p>								
<b>3</b>	<b>Inhalte</b>							
<p>Audiovisual Computing verwendet Methoden und Erkenntnisse aus verschiedenen Bereichen der Informatik, Physik, Mathematik und den Kulturwissenschaften. Das Anwenden solcher Methoden ist wesentlicher Inhalt der Veranstaltung.</p> <p>Folgende Themenbereichen stehen exemplarisch für mögliche Inhalte:</p> <ul style="list-style-type: none"> <li>• Entwicklung von Grundlagentechnologien und Frameworks für interaktive Kunst und Medien</li> <li>• Künstlerische Projekte in der Komposition, Musik, im Bereich Medien und Video</li> <li>• Interaktive Installationen für Messen, Kulturinstitutionen, Museen und Events</li> <li>• Immersive Medien im öffentlichen Raum</li> <li>• Visualisierung und Sonifikation großer Datenmengen</li> <li>• Entwurf von Szenarien und Klangwelten für crossmediale und transmediale Erzählformen</li> <li>• Gestaltung interaktiver Medien (Gaming, Infotainment, Web)</li> <li>• Mathematische Grundlagen der Musik</li> <li>• Physikalische Grundlagen der Musik</li> <li>• Analoge und digitale Klangerzeuger</li> </ul>								

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

	<ul style="list-style-type: none"> <li>• Audiodigitalisierung und Audioformate</li> <li>• MIDI</li> <li>• Virtuelle Instrumente und VST</li> <li>• Digitale Klangbearbeitung und –veränderung</li> <li>• Spezielle Audio-Programmiersprachen</li> <li>• Audio-Bibliotheken für all-purpose Programmiersprachen, insb. C/C++</li> <li>• Agogik und der menschliche Faktor</li> <li>• Die Musik als universell-menschliches Phänomen</li> <li>• Psychoakustik und Musikgenuss</li> <li>• Programmierung von DAWs</li> <li>• Entwicklung von Applikationen und Interfaces für Kunst- und Musikschaffende im professionellen und nichtprofessionellen Einsatz</li> </ul>
<b>4</b>	<b>Teilnahmevoraussetzungen</b> keine
<b>5</b>	<b>Prüfungsgestaltung</b> Klausur oder mündliche Prüfung oder schriftliche Hausarbeit oder Projektarbeit oder Performanzprüfung oder wissenschaftliches Poster oder Kurzpublikationsmanuskript oder Forschungsförderungsantrag oder Praktikums-, Exkursions- oder Tagesprotokoll oder Portfolio oder Lerntagebuch oder Parcourprüfung oder (gem. § 14 (4) RPO) eine Kombination aus verschiedenen Formen von Prüfungsleistungen
<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b> Bestandene Modulprüfung und ggf. Testat
<b>7</b>	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik
<b>8</b>	<b>Modulbeauftragte/r</b> Prof. Dr. Dominic Becking
<b>9</b>	<b>Sonstige Informationen</b> Literaturangaben: <ul style="list-style-type: none"> <li>• Aktuelle Zeitschriften und Proceedings zum Thema.</li> <li>• Steppat, M.: Audioprogrammierung. Hanser, München, 2014.</li> <li>• Boulanger, R., Lazzarini, V. (Hgg.): The Audio Programming Book. MIT Press, Cambridge USA, 2011.</li> <li>• Mazzola, G.: Elemente der Musikinformatik. Birkhäuser, Basel, 2006.</li> <li>• Loy, G.: Musimathics – the mathematical foundations of music, Vol. 1 u. 2. MIT Press, Cambridge USA, 2007.</li> <li>• Gouveia, D.: Getting Started with C++ Audio Programming for Game Development. Packt Publishing, Birmingham, 2013.</li> <li>• Brown, A. R.: Making Music with Java. o.O., 2005</li> <li>• Richard Szeliski (2011): "Computer Vision: Algorithms and Applications", Springer</li> <li>• Gary Bradski, Adrian Kaehler (2008): "Learning OpenCV: Computer Vision with the OpenCV Library", O'Reilly</li> <li>• John F. Hughes, et al. (2014): "Computer Graphics: Principles and Practice", Addison-Wesley.</li> <li>• Dave Shreiner, Graham Sellers, John M. Kessenich, Bill Licea-Kane (2013): "OpenGL Programming Guide: The Official Guide to Learning OpenGL, Version 4.3", Addison-Wesley</li> <li>• Meinhard Müller (2015): Fundamentals of Music Processing: Audio, Analysis, Algorithms, Applications, Springer</li> <li>• Julius O. Smith III (2012): "Physical Audio Signal Processing: for Virtual Musical Instruments and Digital Audio Effects", W3K Publishing</li> <li>• Richard Boulanger, Victor Lazzarini (2010): The Audio Programming Book, MIT Press</li> <li>• John G. Proakis, Dimitris K Manolakis (2014): "Digital Signal Processing", Pearson</li> </ul>

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodul aus Liste 1 „Methoden der Informatik“ Funktionale Programmierung								Kürzel FP
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
5.12	150h	5	5./6. Sem.	halbjährlich bei Nachfrage	SoSe/WS bei Nach- frage	1 Sem.	Wahl- pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>		<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>
	Vorlesung		2 SWS / 30h	45h	Wird in LV bekannt gegeben		60	deutsch
	Praktikum		2 SWS / 30h	45h			15	deutsch
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Die funktionale Programmierung ist neben der objektorientierten Programmierung und der verteilten/nebenläufigen Programmierung ein wichtiges Programmierparadigma. Konzepte aus der funktionalen Programmierung wie beispielsweise Lambda-Ausdrücke halten schrittweise Einzug in moderne Sprachen wie Java9 und C#. Die Veranstaltung führt die Konzepte der funktionalen Programmierung ein und zeigt die Umsetzung in den Beispielsprachen Haskell und Scala. Die Studierenden beherrschen wichtige Konzepte der funktionalen Programmierung und können diese anhand der Sprachen Haskell und Scala anwenden. Sie erkennen funktionale Konzepte in anderen modernen Programmiersprachen und können diese ergebnisorientiert anwenden.							
3	<b>Inhalte</b> Auswahl an Themen zur Wissensvermittlung: <ul style="list-style-type: none"><li>• Funktionen und Operatoren</li><li>• Lambda-Notation</li><li>• Funktionen höherer Ordnung: map, filter, reduce, zip</li><li>• Funktionskomposition und Currying</li><li>• Datenstrukturen</li><li>• (Algebraische) Typen und Typklassen, Polymorphie, Pattern Matching</li><li>• Funktoren und Monaden, Kombinatorbibliotheken</li><li>• Behandlung von optionalen Werten</li><li>• Auswertungsstrategien, Lazyness</li><li>• Modularisierung und Schnittstellen</li><li>• Berechenbarkeit und Lambda-Kalkül</li><li>• Typinferenzsysteme</li><li>• Einführung in die Programmiersprachen Haskell und Scala</li></ul>							
4	<b>Teilnahmevoraussetzungen</b> keine							
5	<b>Prüfungsgestaltung</b> Mündliche Prüfung oder Klausur							
6	<b>Voraussetzung für die Vergabe von Credit Points</b> Testat und bestandene Modulprüfung							
7	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
8	<b>Modulbeauftragte/r</b> Prof. Dr.-Ing. Carsten Gips							
9	<b>Sonstige Informationen</b> <ul style="list-style-type: none"><li>• Pepper, Hofstedt: "Funktionale Programmierung", Springer, 2006</li><li>• Jeuring, Peyton-Jones: "Advanced Functional Programming", Springer, 2009</li><li>• Block, Neumann: "Haskell Intensivkurs", Springer, 2011</li><li>• Lipovaca, M.: "Learn You a Haskell", No Starch Press, 2011</li><li>• Horstmann, C.: "Scala for the Impatient", Addison Wesley, 2012</li><li>• Odersky, M.: "Programming in Scala", Artima, 2011</li></ul>							



**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodul aus Liste 1: Cloud Computing								Kürzel CC
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
5.13	150h	5	5. Sem.	jährlich	WiSe	1 Sem.	Wahl- pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>	
	Seminaristischer Unterricht	2 SWS / 30h	45h	Wird in LV bekannt gegeben		60	deutsch	
	Praktikum / Seminar	2 SWS / 30h	45h			15	deutsch	
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>							
	<ul style="list-style-type: none"> <li>Die Studierenden kennen kommerzielle und Open-Source-Lösungen zur Entwicklung von Cloud-Computing Anwendungen und können diese anwenden.</li> <li>Die Studierenden lernen unter Berücksichtigung verschiedener Performancekriterien, etablierte Algorithmen zur skalierenden Bereitstellung von verteilten Systemressourcen zu beschreiben.</li> <li>Die Studierenden lernen, ein komplexes, verteiltes System zur parallelen Verarbeitung großer Datenmengen zu entwerfen, zu implementieren und einzusetzen.</li> </ul>							
3	<b>Inhalte</b>							
	<ul style="list-style-type: none"> <li>Kommerzielle und Open-Source Cloud-Computing Technologien im Bereich IaaS (z.B. AWS, Azure, Eucalyptus), PaaS (z.B. Heroku, EC2), SaaS (z.B. Cloudgene).</li> <li>Systemarchitekturen, Web- und Data Service-Topologien.</li> <li>Protokolle, Muster und Standards.</li> <li>Big Data Analytics und Parallelisierung (z.B. Hadoop, BigQuery, Storm).</li> <li>Fog / Edge Computing Anwendungen.</li> <li>Resourceverwaltung unter flexiblen Performancekriterien.</li> <li>Stochastische Methoden im Kontext von Load Balancing und Elastic Computing.</li> <li>Monitoring und Anomaly Detection.</li> </ul>							
4	<b>Teilnahmevoraussetzungen</b>							
	Formal: -, Inhaltlich: Kenntnisse aus Datenbanken, Software Engineering, Webbasierte Anwendungen, Verteilte Systeme							
5	<b>Prüfungsgestaltung</b>							
	Projekt-/ Seminararbeit, mündliche Prüfung oder Klausur.							
6	<b>Voraussetzung für die Vergabe von Credit Points</b>							
	Praktikum mit Testat und bestandene Modulprüfung							
7	<b>Verwendung des Moduls (in folgenden Studiengängen):</b>							
	Bachelor Informatik							
8	<b>Modulbeauftragte/r</b>							
	Prof. Dr. Jörg Brunsmann							
9	<b>Sonstige Informationen</b>							



**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodul aus Liste 1 „Methoden der Informatik“ Künstliche Intelligenz								Kürzel KI
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
5.14	150h	5	5./6. Sem.	halbjährlich bei Nachfrage	SoSe/WS bei Nach- frage	1 Sem.	Wahl- pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>		<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>
	Vorlesung	2 SWS / 30h		45h	Wird in LV bekannt gegeben		60	deutsch
	Praktikum	2 SWS / 30h		45h			15	deutsch
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Die Veranstaltung führt in die Prinzipien und grundlegenden Methoden und Verfahren der modernen Künstlichen Intelligenz und deren Anwendung in intelligenten Systemen ein. Mit der erfolgreichen Teilnahme an der Vorlesung und den Praktika werden Kompetenzen zur Auswahl und Anwendung von grundlegenden Methoden der Künstlichen Intelligenz für eine konkrete Problemstellung erworben. Die Teilnehmer sind in der Lage, die erlernten Methoden auf andere Bereiche und Problemstellungen zu übertragen und anzuwenden.							
3	<b>Inhalte</b> Auswahl an Themen zur Wissensvermittlung: <ul style="list-style-type: none"> <li>• Intelligente Agenten</li> <li>• Problemlösen: <ul style="list-style-type: none"> <li>– Informierte und uninformierte Suche (Tiefensuche, Breitensuche, Branch-and-Bound, Best First, A*)</li> <li>– Lokale Suche (Gradientensuche, Simulated Annealing)</li> <li>– Genetische und Evolutionäre Algorithmen</li> <li>– Constraint Satisfaction Problems, Backtracking-Suche mit Heuristiken, Constraint-Propagation und AC3 (Kantenkonsistenz)</li> <li>– Spiele (Minimax Algorithmus, alpha-beta-Pruning, Heuristiken)</li> </ul> </li> <li>• Wissensrepräsentation und Schließen: <ul style="list-style-type: none"> <li>– Aussagenlogik</li> <li>– Prädikatenlogik, Syntax und Semantik, Modelle</li> <li>– Unifikation, Normalform, Resolutionskalkül</li> <li>– Logische Programmierung (Prolog)</li> <li>– Schließen mit Unsicherheit, Bayes'sche Regel, Bayes'sche Netze</li> <li>– Semantische Netze und Ontologien</li> <li>– Planen, Situationskalkül, STRIPS</li> </ul> </li> <li>• Maschinelles Lernen: <ul style="list-style-type: none"> <li>– Entscheidungsbaumverfahren (CAL2, CAL3, ID3, C4.5)</li> <li>– Neuronale Netze: Perzeptron, MLP, Delta-Regel, Backpropagation</li> <li>– Support Vector Machines</li> <li>– (Text-) Klassifikation mit Naive Bayes, Grundlagen der Wahrscheinlichkeitstheorie</li> <li>– Generalisierung, Overfitting, Crossvalidierung, Regularisierung, Boosting</li> <li>– Reinforcement Learning</li> <li>– Unüberwachtes Lernen: RBF, kNN</li> </ul> </li> <li>• Unschärfe Textsuche, Ähnlichkeiten zw. Texten</li> <li>• Empfehlungssysteme</li> <li>• Autonome mobile Systeme</li> <li>• Wechselnde Inhalte der Praktika zu aktuellen Themen</li> </ul>							
4	<b>Teilnahmevoraussetzungen</b> keine							
5	<b>Prüfungsgestaltung</b> Mündliche Prüfung oder Klausur							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b> Testat und bestandene Modulprüfung
<b>7</b>	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik
<b>8</b>	<b>Modulbeauftragte/r</b> Prof. Dr.-Ing. Carsten Gips
<b>9</b>	<b>Sonstige Informationen</b> <ul style="list-style-type: none"> <li>• Russel, S., Norvig, P: "Artificial Intelligence. A Modern Approach", Prentice Hall, 2014</li> <li>• Ertel, W.: "Grundkurs Künstliche Intelligenz", Springer Vieweg, 2016</li> <li>• Bishop, C.: "Pattern Recognition and Machine Learning", Springer, 2007</li> <li>• Witten et al.: "Data Mining: Practical Machine Learning Tools and Techniques", Morgan Kaufmann, 2011</li> <li>• Mitchell: "Machine Learning", Mcgraw-Hill Education, 1997</li> </ul>

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodul aus Liste 1 „Methoden der Informatik“ Methoden der Computergrafik, Schwerpunkt algorithmische Geometrie und geometrische Modellierung								Kürzel GL1
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
5.15	150h	5	5./6. Sem.	halbjährlich bei Nachfrage	SoSe/ WS bei Nachfrage	1 Sem.	Wahl- pflicht	B.Sc.
<b>1</b>	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>	
	Vorlesung	2 SWS / 30h	45h	Wird in LV bekannt		60		
	Praktikum / Seminar	2 SWS / 30h	45h	Gegeben.		15	deutsch	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Aufbauend auf den erworbenen Grundlagen der Vorlesung Computergrafik können die Studierenden Methoden aus Teilen der Bereiche algorithmische Geometrie und geometrische Modellierung anwenden. Die Studierenden können die gestellten Probleme analysieren und das passende Verfahren aus dem Gebiet der Computergrafik auswählen							
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Algorithmen und Datenstrukturen: Verfahren aus der algorithmische Geometrie.</li> <li>• Geometrische Modellierung.</li> </ul>							
<b>4</b>	<b>Teilnahmevoraussetzungen</b> Formal: - Inhaltlich: Kenntnisse aus Modul Computergrafik (CG)							
<b>5</b>	<b>Prüfungsgestaltung</b> Mündliche Prüfung oder Klausur							
<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b> Teilnahme am Praktikum mit Testat, bestandene Modulprüfung							
<b>7</b>	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
<b>8</b>	<b>Modulbeauftragte/r</b> Prof. Dr. Kerstin Müller							
<b>9</b>	<b>Sonstige Informationen</b> <ul style="list-style-type: none"> <li>• Gerald Farin: Curves and Surfaces for CAGD: A Practical Guide, Morgan Kaufmann</li> <li>• de Berg, M., Cheong, O., van Kreveld, M., Overmars, M.: Computational Geometry -Algorithms and Applications, Springer</li> </ul>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodul aus Liste 1 „Methoden der Informatik“ Methoden der Computergrafik, Schwerpunkt Bildgenerierung								Kürzel VL1
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
5.16	150h	5	5./6. Sem.	halbjährlich bei Nachfrage	SoSe/WS bei Nachfrage	1 Sem.	Wahl- pflicht	B.Sc.
<b>1</b>	<b>Lehrveranstaltungs- art</b>		<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>
	Vorlesung		2 SWS / 30h	45h	Wird in LV bekannt gegeben.		60	deutsch
	Praktikum / Seminar		2 SWS / 30h	45h			15	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Aufbauend auf den erworbenen Grundlagen der Vorlesung Computergrafik können die Studierenden Methoden aus Teilen der Bereiche Visualisierung und globale Beleuchtung anwenden. Die Studierenden können die gestellten Probleme analysieren und das passende Verfahren aus dem Gebiet der Computergrafik auswählen.							
<b>3</b>	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Globale Beleuchtungsmethoden: Raytracing, Radiosity.</li> <li>• Visualisierung.</li> </ul>							
<b>4</b>	<b>Teilnahmevoraussetzungen</b> Formal: - Inhaltlich: Kenntnisse aus Modul Computergrafik (CG)							
<b>5</b>	<b>Prüfungsgestaltung</b> Mündliche Prüfung oder Klausur							
<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b> Teilnahme am Praktikum mit Testat, bestandene Modulprüfung							
<b>7</b>	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
<b>8</b>	<b>Modulbeauftragte/r</b> Prof. Dr. Kerstin Müller							
<b>9</b>	<b>Sonstige Informationen</b> <ul style="list-style-type: none"> <li>• Hearn D., Baker M.P.: Computer Graphics with OpenGL, Pearson International Edition.</li> <li>• Foley J., van Dam A., Fisher S., Hughes J.: Computer Graphics – Principles and Practice, Addison-Wesley</li> </ul>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodul aus Liste 1 „Methoden der Informatik“								Kürzel PIS
Praktische Aspekte der IT-Sicherheit								
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
5.18	150 h	5	6. Sem.	Bei Nachfrage jährlich	SoSe	1 Sem.	Wahl- pflicht	B.Sc.
<b>1</b>	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>		<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>
	Vorlesung	2 SWS / 30 h		45 h	Wird in LV bekannt gegeben		60	Deutsch
	Praktikum / Seminar	2 SWS/ 30 h		45 h			15	Deutsch
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>							
	<p>Fachkompetenz: Die Studierenden haben ein vertieftes Verständnis für die Vorgehensweise von Angreifern gegen IT-Systeme und -Netze und von konkreten Angriffen und Gefahren im Internet. Sie sind in die Lage, Schutzmaßnahmen zu beurteilen sowie bei der Implementierung solcher Schutzmaßnahmen mitwirken zu können.</p> <p>Methodenkompetenz: Die Studierenden können Angriffe erkennen, die Phasen eines Angriffs beschreiben, strukturieren und klassifizieren sowie geeignete Schutzmaßnahmen skizzieren und anwenden. Zudem können Studierende die Eignung von (Schutz-)Maßnahmen bewerten und die Maßnahmen anwenden.</p> <p>Sozialkompetenz: Aufgrund der Teamarbeit, unter anderem bei den praktischen Aufgaben, sind die Studierenden fähig, Lösungswege in der Gruppe zu entwickeln und Aufgaben kooperativ zu lösen.</p>							
<b>3</b>	<b>Inhalte</b>							
	<ul style="list-style-type: none"> <li>- Vorgehensmodelle von Angreifern</li> <li>- Angriffsverfahren (lokal/entfernt)</li> <li>- Schutzmaßnahmen</li> <li>- Typische Angriffe auf Systeme <ul style="list-style-type: none"> <li>o Angriffe auf Schwächen in Protokollen</li> <li>o Angriffe auf die Konfiguration von Systemen</li> <li>o Angriffe auf Webanwendungen</li> <li>o Spezielle Überwachungs-, bzw. Angriffsprogramme</li> </ul> </li> <li>- Angriffserkennung</li> <li>- Intrusion Detection Systeme</li> </ul>							
<b>4</b>	<b>Teilnahmevoraussetzungen</b>							
	<p>Formal: -</p> <p>Inhaltlich: Grundlegende Kenntnisse aus Mathematik für Informatiker I und II, Programmierkenntnisse, Technische Informatik, Betriebssysteme, Datenbanken, Webtechnologien</p>							
<b>5</b>	<b>Prüfungsgestaltung</b>							
	Mündliche Prüfung oder Projektarbeit							
<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b>							
	Bestehen der Modulprüfung. Es gilt § 17 „Fortschrittsregelung“ der Prüfungsordnung für den Bachelorstudiengang Informatik..							
<b>7</b>	<b>Verwendung des Moduls (in folgenden Studiengängen):</b>							
	Bachelorstudiengang Informatik							
<b>8</b>	<b>Modulbeauftragte/r</b>							
	Prof. Dr. Christoph Thiel							
<b>9</b>	<b>Sonstige Informationen</b>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodulaus Liste 1 „Methoden der Informatik“								Kürzel BSZ
Sicherheit und Zuverlässigkeit								
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
5.19	150 h	5	5. Sem.	Bei Nachfrage jährlich	WiSe	1 Sem.	Wahl- pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>	<b>gepl. Gruppengr.</b>	<b>Sprache</b>		
	Vorlesung	2 SWS / 30 h	45 h	Wird in LV bekannt gegeben	60	Deutsch		
	Praktikum / Seminar	2 SWS/ 30 h	45 h		15	Deutsch		
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>							
	<p>Fachkompetenz: Die Studierenden verstehen die Zusammenhänge und Unterschiede der Sicherheitsbegriffe Dependability, Safety und Security. Sie sind mit typischen Schwachstellen und Bedrohungen vertraut und kennen geeignete Gegenmaßnahmen und –mechanismen zur Erhöhung der Zuverlässigkeit und Sicherheit.</p> <p>Methodenkompetenz: Die Studierenden können erste Einschätzungen der Sicherheit und Zuverlässigkeit von Systemen und Software vornehmen, Möglichkeiten und Grenzen von Lösungen bewerten und mögliche Verbesserungen vorschlagen.</p> <p>Sozialkompetenz: Aufgrund der Teamarbeit, unter anderem bei den praktischen Aufgaben, sind die Studierenden fähig, Lösungswege in der Gruppe zu entwickeln und Aufgaben kooperativ zu lösen.</p>							
3	<b>Inhalte</b>							
	<ul style="list-style-type: none"> <li>- Grundlegende Begriffe und Probleme der Sicherheit und der Zuverlässigkeit;</li> <li>- Ziele der Verlässlichkeit und Sicherheit (Vertraulichkeit, Integrität, Verfügbarkeit, Safety, Wartbarkeit, ...);</li> <li>- Schwachstellen-, Bedrohungs-, Impact- und Risikoanalysen;</li> <li>- Grundlegende Formen der Behandlung von Risiken</li> <li>- Aufbau und Struktur von Risikobehandlungsplänen</li> <li>- Maßnahmen und Mechanismen zur Erhöhung der Zuverlässigkeit und Sicherheit von Software und Systemen (Kryptographie, Authentifizierung, Zugriffskontrolle, Protokolle, Firewalls, etc.)</li> </ul>							
4	<b>Teilnahmevoraussetzungen</b>							
	<p>Formal: -</p> <p>Inhaltlich: Kenntnisse aus Mathematik für Informatiker I und II, Programmierkenntnisse, Technische Informatik, Betriebssysteme, Datenbanken, Webtechnologien</p>							
5	<b>Prüfungsgestaltung</b>							
	Mündliche Prüfung oder Klausur oder Projektarbeit							
6	<b>Voraussetzung für die Vergabe von Credit Points</b>							
	<p>Bestehen der Modulprüfung.</p> <p>Es gilt § 17 „Fortschrittsregelung“ der Prüfungsordnung für den Bachelorstudiengang Informatik..</p>							
7	<b>Verwendung des Moduls (in folgenden Studiengängen):</b>							
	Bachelorstudiengang Informatik							
8	<b>Modulbeauftragte/r</b>							
	Prof. Dr. Christoph Thiel							
9	<b>Sonstige Informationen</b>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodul aus Liste 1 „Methoden der Informatik“ Spezielle Methoden der Programmierung								Kürzel SM
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
5.20	150h	5	5./6. Sem.	halbjährlich bei Nachfrage	SoSe/WS bei Nach- frage	1 Sem.	Wahl- pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>		<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>
	Vorlesung		2 SWS / 30h	45h	Wird in LV bekannt gegeben		60	deutsch
Praktikum		2 SWS / 30h	45h			15	deutsch	
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Die Teilnehmer erhalten einen Überblick über verschiedene Programmierparadigmen und Sprachkonzepte sowie ihre Anwendungsgebiete. Sie lernen ausgewählte Konzepte und deren Umsetzung in modernen Programmiersprachen genauer kennen und verstehen Aufbau und Prinzipien von Programmiersprachen. Die Teilnehmer werden befähigt, selbständig neue Sprachkonstrukte zu erlernen und einzuordnen. Nach erfolgreichem Abschluss des Moduls sind sie in der Lage, problem- und lösungsorientiert ein geeignetes Paradigma und eine entsprechende Programmiersprache auszuwählen und einzusetzen.							
3	<b>Inhalte</b> Auswahl an Themen zur Wissensvermittlung: <ul style="list-style-type: none"> <li>• Überblick über Programmierparadigmen und -konzepte</li> <li>• Vertiefung ausgewählter Paradigmen <ul style="list-style-type: none"> <li>– objektorientierte Programmierung (Ruby: Duck Typing, offene Klassen, Mixins, Metaprogrammierung)</li> <li>– funktionale Programmierung (Haskell: Lambda-Notation, Currying, Funktionen höherer Ordnung, Datentypen/Typklassen, Monaden)</li> <li>– hybride objektorientierte und funktionale Programmierung (Scala: Listcomprehensions, Traits, Companion-Objekte, Pattern-Matching, Option-Datentyp)</li> <li>– logische Programmierung (Prolog: Unifikation, Resolution, Rekursion, Listen: Head, Tail, Akkumulatoren; Cuts)</li> <li>– nebenläufige/parallele Programmierung (beispielsweise Erlang)</li> </ul> </li> <li>• Vertiefung ausgewählter Konzepte <ul style="list-style-type: none"> <li>– Berechenbarkeit und Lambda-Kalkül</li> <li>– Auswertungsstrategien, Lazyness</li> <li>– Algebraische Typen mit Pattern Matching</li> <li>– Kombinatorbibliotheken, Funktoren und Monaden</li> <li>– Nebenläufigkeit auf der JVM mittels Aktoren in Akka</li> <li>– Behandlung von optionalen oder null-/nil-Werten</li> </ul> </li> <li>• DSL und Tools (Xtext/Xtend, Eclipse Plugins, Antlr)</li> <li>• Modellbasierte Entwicklung (UML; Eclipse EMF/GMF, ...)</li> <li>• Modellbasierte Entwicklung (Matlab/Simulink)</li> </ul>							
4	<b>Teilnahmevoraussetzungen</b> keine							
5	<b>Prüfungsgestaltung</b> Mündliche Prüfung oder Klausur							
6	<b>Voraussetzung für die Vergabe von Credit Points</b> Testat und bestandene Modulprüfung							
7	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

<b>8</b>	<b>Modulbeauftragte/r</b> Prof. Dr.-Ing. Carsten Gips
<b>9</b>	<b>Sonstige Informationen</b> <ul style="list-style-type: none"><li>• Tate, B.A.: "Seven Languages in Seven Weeks", Pragmatic Bookshelf Inc., 2010</li><li>• Scott, M.L.: "Programming Language Pragmatics", Morgan Kaufmann, 2009</li><li>• Lipovaca, M.: "Learn You a Haskell", No Starch Press, 2011</li><li>• Block, Neumann: "Haskell Intensivkurs", Springer, 2011</li><li>• Horstmann, C.: "Scala for the Impatient", Addison Wesley, 2012</li><li>• Odersky, M.: "Programming in Scala", Artima, 2011</li><li>• Subramaniam, V.: "Programming Groovy 2", O'Reilly, 2013</li><li>• Thomas, Hunt: "Programming Ruby", O'Reilly, 2013</li><li>• Voelter, M.: "DSL Engineering: Designing, Implementing and Using Domain-Specific Languages", CreateSpace Independent Publishing Platform, 2013</li><li>• Bettini, L.: "Implementing Domain-Specific Languages with Xtext and Xtend", PACKT Publishing, 2013</li><li>• Pepper, Hofstedt: "Funktionale Programmierung", Springer, 2006</li><li>• Johan Jeuring, Simon Peyton Jones: "Advanced Functional Programming", Springer, 2009</li></ul>



**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodul aus Liste 2 „Anwendungen der Informatik“ Anwendungen der Computergrafik, Schwerpunkt CAGD und effiziente Datenstrukturen								Kürzel CG2
Nr.	Workload	Credit Points	Studien-semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
6.20	450h	15	5./6. Sem.	halbjährlich bei Nachfrage	SoSe/WS bei Nachfrage	1 Sem.	Wahl- pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>	
	Vorlesung	2 SWS / 30h	45h	Wird in LV bekannt gegeben		60		
	Praktikum / Seminar	4 SWS / 30h	315h			15	deutsch	
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Aufbauend auf den erworbenen Grundlagen der Vorlesung Computergrafik bearbeiten die Studierenden aktuelle Problemstellungen aus der Computergrafik und sind in der Lage geeignete Methoden zu evaluieren und aktuelle Werkzeuge aus der Computergrafik auszuwählen.							
3	<b>Inhalte</b> Die Themen kommen aus aktuellen Forschungsgebieten und Forschungskooperationen mit Industriepartnern und bewegen sich in folgenden Gebieten: <b>CAGDL2</b> <ul style="list-style-type: none"> <li>• Effiziente Datenstrukturen der Computergrafik.</li> <li>• Computer Aided Geometric Design</li> </ul> Die Aufgaben werden im Team bearbeitet, interdisziplinäre Projekte sind ebenfalls im Rahmen dieses Wahlpflichtmoduls möglich.							
4	<b>Teilnahmevoraussetzungen</b> Formal: - Inhaltlich: Kenntnisse aus Modul Computergrafik (CG)							
5	<b>Prüfungsgestaltung</b> Projektarbeit							
6	<b>Voraussetzung für die Vergabe von Credit Points</b> Erfolgreicher Abschluss der Projektarbeit							
7	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
8	<b>Modulbeauftragte/r</b> Prof. Dr. Kerstin Müller							
9	<b>Sonstige Informationen</b> <ul style="list-style-type: none"> <li>• Gerald Farin: Curves and Surfaces for CAGD: A Practical Guide, Morgan Kaufmann</li> <li>• de Berg, M., Cheong, O., van Kreveld, M., Overmars, M.: Computational Geometry - Algorithms and Applications, Springer</li> </ul>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodul aus Liste 2 „Anwendungen der Informatik“ Anwendungen der Computergrafik, Schwerpunkt Virtual Reality und Visualisierungstechniken								Kürzel VR2
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
6.21	450h	15	5./6. Sem.	halbjährlich bbei Nachfrage	SoSe/WS bei Nachfrage	1 Sem.	Wahl- pflicht	B.Sc.
<b>1</b>	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>	
	Vorlesung	2 SWS / 30h	45h	Wird in LV bekannt gegeben.		60		
	Praktikum / Seminar	4 SWS / 30h	315			15	deutsch	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Aufbauend auf den erworbenen Grundlagen der Vorlesung Computergrafik bearbeiten die Studierenden aktuelle Problemstellungen aus der Computergrafik und sind in der Lage geeignete Methoden zu evaluieren und aktuelle Werkzeuge aus der Computergrafik auszuwählen							
<b>3</b>	<b>Inhalte</b> Die Themen kommen aus aktuellen Forschungsgebieten und Forschungskooperationen mit Industriepartnern und bewegen sich in folgenden Gebieten: <ul style="list-style-type: none"> <li>• Visualisierungstechniken: Methoden zur optimierten Darstellung.</li> <li>• Useability und Informationsvisualisierung.</li> <li>• (serious) Games.</li> <li>• Virtual Reality Anwendungen.</li> </ul> Die Aufgaben werden im Team bearbeitet, interdisziplinäre Projekte sind ebenfalls im Rahmen dieses Wahlpflichtmoduls möglich.							
<b>4</b>	<b>Teilnahmevoraussetzungen</b> Formal: - Inhaltlich: Kenntnisse aus Modul Computergrafik (CG)							
<b>5</b>	<b>Prüfungsgestaltung</b> Projektarbeit							
<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b> Erfolgreicher Abschluss der Projektarbeit							
<b>7</b>	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
<b>8</b>	<b>Modulbeauftragte/r</b> Prof. Dr. Kerstin Müller							
<b>9</b>	<b>Sonstige Informationen</b> <ul style="list-style-type: none"> <li>• Hearn D., Baker M.P.: Computer Graphics with OpenGL, Pearson International Edition.</li> <li>• Foley J., van Dam A., Feiner S., Hughes J.: Computer Graphics – Principles and Practice, Addison-Wesley</li> </ul>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

<b>Wahlpflichtmodul aus Liste 2 „Anwendungen der Informatik“ Computerkunst, Sozioinformatik und neue Kommunikationsalgorithmen</b>								<b>Kürzel CSK</b>
<b>Nr.</b>	<b>Workload</b>	<b>Credit Points</b>	<b>Studien- semester</b>	<b>Häufigkeit</b>	<b>Sem.</b>	<b>Dauer</b>	<b>Art</b>	<b>Q-Niveau</b>
<b>6.22</b>	<b>450h</b>	<b>15</b>	5./6. Sem.	halbjährlich bei Nachfrage	SoSe/WS bei Nachfrage	1 Sem.	Wahl- pflicht	B.Sc.
<b>1</b>	<b>Lehrveranstaltungs- art</b>		<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>
	Vorlesung		2 SWS / 30h	45h	Wird in LV bekannt gegeben.		60	
	Praktikum / Seminar		4 SWS / 30h	315h			15	deutsch
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Die Studierenden haben einen Überblick über aktuelle Techniken und Anwendungen der Computerkunst. Sie entwerfen und implementieren und/oder integrieren Interaktionstechniken und Hardware der Wechselwirkung zwischen sozialen Gruppen und Softwaresystemen und werden vertraut mit Gestaltungsprinzipien und Realisierungsmethoden für Software auch mit großen Nutzergruppen. Sie konstruieren neue Kommunikationsformen und sind vertraut mit entsprechenden neuen Algorithmen.							
<b>3</b>	<b>Inhalte</b> Die zu erstellenden Softwareapplikationen kommen aus den Anwendungsfeldern Medieninformatik, Visualisierung, Computer Vision, Musikinformatik, KI, Spieltheorie, Robotik, Kunst, Psychologie und Soziologie. Eine interdisziplinäre Ausrichtung ist erwünscht. Kreative, neue Kommunikationsformen, Handlungsszenarien und Interaktionsmechanismen von Mensch und Maschine bzw. von Mensch zu Mensch mit Kommunikationspartner Maschine sollen konzipiert und umgesetzt werden. Aspekte großer Skalierung und Masseninteraktion können einbezogen werden.							
<b>4</b>	<b>Teilnahmevoraussetzungen</b> keine							
<b>5</b>	<b>Prüfungsgestaltung</b> Projektarbeit							
<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b> Erfolgreicher Abschluss der Projektarbeit							
<b>7</b>	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
<b>8</b>	<b>Modulbeauftragte/r</b> Prof. Dr. Dominic Becking, Prof. Dr. Kerstin Müller							
<b>9</b>	<b>Sonstige Informationen</b>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodul aus Liste 2 „Anwendungen der Informatik“ Datenbankanwendungen								Kürzel DBA
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
6.23	450	15	5./6. Sem.	halbjährlich bei Nachfrage	bei Nachfrage	1 Sem.	Wahl- pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>		<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>
	Vorlesung		2 SWS / 30h	360h	Wird in LV bekannt gegeben.		60	deutsch
	Praktikum		4 SWS / 60h				15	deutsch
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Datenbanken sind die Grundlage jedes großen Softwaresystems. Das Zusammenwirken von Programmierung und Datenbank ist dabei von größter Wichtigkeit. Die Studierenden erheben und formulieren die Anforderungen verschiedener Softwaresystemklassen an Datenbanken. Sie entwerfen spezielle Datenmodelle für unterschiedliche Systemklassen und implementieren diese. Sie entwerfen und realisieren ein Softwaresystem bestehend aus Anwendungsprogramm und darauf abgestimmter Datenbank. Sie implementieren Programmlogik in der Datenbank mit Persistent Stored Modules. Die Teilnehmer verwenden verschiedene APIs zur Anbindung von Anwenderprogrammen an Datenbanken..							
3	<b>Inhalte</b> Folgende Themenbereichen stehen exemplarisch für mögliche Inhalte: <ul style="list-style-type: none"> <li>• Softwaresysteme und Anforderungen an Datenbanken</li> <li>• Anforderungsanalyse für Datenbankanwendungen</li> <li>• Objektorientierte und ER-Modellierung</li> <li>• Persistent Stored Modules (SQL/PSM)</li> <li>• Trigger</li> <li>• Impedance Mismatch</li> <li>• Cursors</li> <li>• Objektrelationale Mappings – Object Language Bindings (SQL/OLB)</li> <li>• APIs und Frameworks für Datenbankzugriffe</li> </ul>							
4	<b>Teilnahmevoraussetzungen</b> Formal: - Inhaltlich: Inhalte des Moduls 3.2 „Datenbanken I“ (DB1)							
5	<b>Prüfungsgestaltung</b> Performanzprüfung oder mündliche Prüfung oder Hausarbeit oder Projektarbeit oder Praktikums-, Exkursions- oder Tagesprotokoll oder Portfolio oder Lerntagebuch oder Parcourprüfung							
6	<b>Voraussetzung für die Vergabe von Credit Points</b> Praktikum mit Testat Bestandene Modulprüfung							
7	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
8	<b>Modulbeauftragte/r</b> Prof. Dr. Dominic Becking							
9	<b>Sonstige Informationen</b>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodul aus Liste 2 „Anwendungen der Informatik“ Embedded Software								Kürzel ESW
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
6.24	450h	15	5./6. Sem.	halbjährlich bei Nachfrage	SoSe/WS bei Nach- frage	1 Sem.	Wahl- pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>	
	Vorlesung	2 SWS / 30h	45h	Wird in LV bekannt gegeben.		60	deutsch	
	Praktikum	4 SWS / 60h	315h			15	deutsch	
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Die Studierenden wissen die Methoden des Softwareengineerings für Embedded Systems anzuwenden unter Berücksichtigung der Randbedingungen dieser Systeme. Sie besitzen die Fähigkeit, die Entwicklung von Software für Embedded Systems von der Planung bis zur Realisierung durchzuführen. Zur Vertiefung erarbeiten die Studierenden eigenständig Projekte, in denen sie „Embedded Software“ für eine Anwendung entwickeln. In der teamorientierten Projektarbeit werden die Projektmanagement- und Selbst-Kompetenzen vertieft.							
3	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Anforderungen / Requirements</li> <li>• Software Architektur und Design Patterns</li> <li>• UML/SysML für Embedded Systems und Modellbasierte Entwicklung</li> <li>• Embedded Betriebssysteme</li> <li>• Qualitätssicherung und Normen</li> <li>• Wechselnde Inhalte der Projekte zu aktuellen Themen, z.B. <ul style="list-style-type: none"> <li>◦ Internet-of-Things-Technologien</li> </ul> </li> </ul> <p>Als vertiefendes Projekt wird beispielsweise eine Anwendung entwickelt (z.B. für das Internet der Dinge, Computer Vision, Robotik, Mess- und Regelungstechnik). Dabei können ggf. Hardware-Abstraction-Layer oder Betriebssysteme für eingebettete Systeme (z.B. Embedded Linux, RTOS) eingesetzt werden.</p>							
4	<b>Teilnahmevoraussetzungen</b> <b>Formal:</b> - <b>Inhaltlich:</b> Kenntnisse eingebetteter Systeme, Software Engineering und C++-Programmierung							
5	<b>Prüfungsgestaltung</b> Projektarbeit							
6	<b>Voraussetzung für die Vergabe von Credit Points</b> Erfolgreicher Abschluss der Projektarbeit							
7	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
8	<b>Modulbeauftragte/r</b> Prof. Dr. Dr.-Ing. Matthias König							
9	<b>Sonstige Informationen</b> Literatur wird in der Veranstaltung bekannt gegeben.							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodul aus Liste 2 „Anwendungen der Informatik“ Internetsicherheit								Kürzel BIS
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
6.25	450 h	15	5. Sem.	Bei Nachfrage jährlich	WiSe	1 Sem.	Wahl- pflicht	B.Sc.
<b>1</b>	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>	<b>gepl. Gruppengr.</b>	<b>Sprache</b>		
	Vorlesung	2 SWS / 30 h	60 h	Wird in LV bekannt gegeben.	60	Deutsch		
	Praktikum / Seminar	4 SWS/ 60 h	300 h		15	Deutsch		
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>							
	<p>Fachkompetenz: Die Studierenden kennen die wichtigsten Basistechnologien zur Absicherung von Netzen unterscheiden. Sie weisen ein tiefgehendes Verständnis von Sicherheitsmechanismen auf den unterschiedlichen Protokollschichten auf (Anwendungsschicht, Transportschicht, Vermittlungsschicht, Sicherungsschicht, physikalische Schicht) und kennen den Aufbau, die Prinzipien, die Architektur und Funktionsweise von Sicherheitskomponenten und –systemen im Bereich der Internetsicherheit. Sie sind in der Lage, die Charakteristiken und Grundprinzipien des Problemraumes Internetsicherheit detailliert zu erläutern und weisen auf diesem Feld ein fundiertes Wissen in Praxis und Theorie auf.</p> <p>Darüber hinaus kennen sie aktuelle Entwicklungen im Bereich Internetsicherheit und können diese erläutern (z.B. Sicherheit in peer-to-peer Systemen, Sicherheit in mobilen Netzen, Sicherheit im Cloud-Computing, Block Chains, etc.).</p> <p>Methodenkompetenz: Die Studierenden können die Grundlagen der IT Sicherheit sowie der Kryptographie auf den Bereich Kommunikationsnetze anwenden und somit Lösungen für Internetsicherheit entwickeln und bewerten. .</p> <p>Sozialkompetenz: Aufgrund der Teamarbeit, unter anderem bei den praktischen Aufgaben, sind die Studierenden fähig, Lösungswege in der Gruppe zu entwickeln und Aufgaben kooperativ zu lösen.</p>							
<b>3</b>	<b>Inhalte</b>							
	<ul style="list-style-type: none"> <li>- Internetsicherheit: Einführung, Motivation und Herausforderungen</li> <li>- Grundlagen: Referenzmodell für Netzsicherheit, Sicherheitsstandards für Netze und das Internet, Bedrohungen, Angriffe, Sicherheitsdienste und -mechanismen</li> <li>- Kryptographische Grundlagen zur Absicherung von Netzen: symmetrische Kryptographie und asymmetrische Kryptographie und deren Anwendung in Netzen, unterstützende Mechanismen zur Implementierung von Sicherheitslösungen, Public Key Infrastrukturen</li> <li>- Sicherheit auf den unterschiedlichen Protokollschichten (Anwendungsschicht, Transportschicht, Vermittlungsschicht, Sicherungsschicht, physikalische Schicht)</li> <li>- Angewandte Internetsicherheit: Firewalls, Intrusion Detection Systems, Identity Management</li> <li>- Ausgewählte Themen der Internetsicherheit: Sicherheit für verteilte Systeme, Sicherheit für Webanwendungen und Webservices, Sicherheit für Cloud-Computing</li> <li>- Wechselnde Inhalte der Projekte zu aktuellen Themen</li> <li>- Nachbildung von Angriffsszenarien und Gegenmaßnahmen im Labor</li> </ul>							
<b>4</b>	<b>Teilnahmevoraussetzungen</b>							
	<p>Formal: -</p> <p>Inhaltlich: Kenntnisse der Java- oder C++-Programmierung, Grundkenntnisse der Technischen Informatik, Verteilte Systeme und Kommunikationsnetze</p>							
<b>5</b>	<b>Prüfungsgestaltung</b>							
	Mündliche Prüfung oder Klausur oder Projektarbeit							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b> Bestehen der Modulprüfung. Es gilt § 17 „Fortschrittsregelung“ der Prüfungsordnung für den Bachelorstudiengang Informatik.
<b>7</b>	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik
<b>8</b>	<b>Modulbeauftragte/r</b> Prof. Dr. Christoph Thiel
<b>9</b>	<b>Sonstige Informationen</b> Literatur: <ul style="list-style-type: none"> <li>• Eckert, C.: IT-Sicherheit: Konzepte – Verfahren, Oldenbourg Wissenschaftlicher Ver-lag; ISBN: 978-3-486-72138-6, 8. Auflage 2013.</li> <li>• Schwenk, Jörg: Sicherheit und Kryptographie im Internet: Von Sicherer E-Mail bis zu IP-Verschlüsselung (German Edition), Vieweg+Teubner Verlag ISBN: 978-3834808141 3. Aufl. 2010</li> <li>• Stallings ,William; Network Security Essentials, 4th Edition, Prentice Hall, ISBN: 978-0-136-10805-9, 2010,</li> <li>• Aktuelle Fachartikel</li> </ul>

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodul aus Liste 2 „Anwendungen der Informatik“								Kürzel MOB
Mobile Applikationen								
Nr.	Workload	Credit Points	Studien-semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
6.26	450h	15	5./6. Sem.	halbjährlich	SoSe/WiSe	1 Sem.	Wahl-pflicht	B.Sc.
1	<b>Lehrveranstaltungs-art</b>		<b>Kontaktzeit</b>	<b>Selbst-studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>
	Vorlesung		2 SWS / 30h	60h	Wird in LV bekannt gegeben.		60	deutsch
Praktikum / Seminar		4 SWS / 60h	300h			15	deutsch	
2	<p><b>Lernergebnisse (learning outcomes) / Kompetenzen</b></p> <p>Softwareentwicklung für mobile Geräte ist die Weiterentwicklung der traditionellen (Anwendungs-)Softwareentwicklung hin zu einer offenen, verteilten und dynamischen Technologie, die mobile, leistungsfähige Geräte mit traditionellen Strukturen des Internets und verteilten Softwaresystemen verbindet.</p> <p>Ziel dieses Moduls ist es, dass die Studierenden</p> <ul style="list-style-type: none"> <li>• die Besonderheiten und Randbedingungen sowie Konzepte und Lösungsansätze der mobilen Applikationsentwicklung verstehen und auch quantitativ abschätzen können,</li> <li>• darauf zugeschnittene Lösungen der Systemarchitektur und der Anwendungsentwicklung kennen lernen,</li> <li>• und diese Kenntnisse zur Lösung von konkreten Aufgabenstellungen praktisch anwenden können.</li> </ul> <p>Die Studierenden erhalten einen Überblick über aktuelle Hardwareplattformen und mobile Betriebssysteme und lernen ein aktuelles System (beispielsweise Android) zu beherrschen. Nach dem Besuch der Veranstaltung sind sie in der Lage, unter Berücksichtigung von Usability-, Energie- und Sicherheitsaspekten eigene mobile Anwendungen zu entwickeln.</p> <p>Zur Vertiefung erarbeiten die Studierenden eigenständig Projekte und vertiefen während der teamorientierten Projektarbeit die Projektmanagement- und Selbst-Kompetenzen.</p>							
3	<p><b>Inhalte</b></p> <ul style="list-style-type: none"> <li>- Technologie mobiler Anwendungen, Hardware, Betriebssysteme</li> <li>- Anwendungen &amp; Anwendungsentwicklung</li> <li>- Aktuelle Software-Frameworks für mobile Anwendungen</li> <li>- Nutzung von Hardwarebausteinen mobiler Geräte</li> <li>- Einbindung vorhandener Sensoren und Schnittstellen</li> <li>- Wechselnde Inhalte der Projekte zu aktuellen Themen, z.B. <ul style="list-style-type: none"> <li>- Ressourcenmanagement in mobilen Systemen und Energieaspekte</li> <li>- Sicherheit für mobile Systeme und Anwendungen</li> </ul> </li> </ul>							
4	<p><b>Teilnahmevoraussetzungen</b></p> <p>Formal: -, Inhaltlich: Kenntnisse der Programmierung, Software Engineering</p>							
5	<p><b>Prüfungsgestaltung</b></p> <p>Projektarbeit</p>							
6	<p><b>Voraussetzung für die Vergabe von Credit Points</b></p> <p>Erfolgreicher Abschluss der Projektarbeit</p>							
7	<p><b>Verwendung des Moduls (in folgenden Studiengängen):</b></p> <p>Bachelor Informatik</p>							
8	<p><b>Modulbeauftragte/r</b></p> <p>Prof. Dr. Martin Hoffmann (stellvertretend Prof. Dr. Dr.-Ing. Matthias König)</p>							
9	<p><b>Sonstige Informationen</b></p> <p>Literaturhinweise: Thomas Künneth: Android 7 - Das Praxisbuch für Entwickler, Rheinwerk Verlag 2017</p>							



**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodul aus Liste 2: Full Stack Development								Kürzel FSD
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
6.27	450h	15	6. Sem.	jährlich	SoSe	1 Sem.	Wahl- pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>		<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>
	Seminaristischer Unterricht Praktikum / Seminar		2 SWS / 30h 4 SWS / 60h	60h 300h	Wird in LV bekannt gegeben.		60 15	deutsch deutsch
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>							
	<ul style="list-style-type: none"> <li>Die Studierenden lernen aktuelle Architekturen, Technologien und Werkzeuge zur Entwicklung von Full Stack Anwendungen für unterschiedliche Endgeräte kennen und können diese anwenden.</li> <li>Die Studierenden lernen, für praxisnahe Problemstellungen eigenständig geeignete Lösungen zu finden und diese im praktischen Umgang mit Softwarewerkzeugen zielgerichtet zu implementieren und zu testen.</li> <li>Die Studierenden lernen, anwendungsfall-spezifische Technologie-Entscheidungen für Frontend-, Backend- und Datenbankkomponenten verteilter Anwendungen zu treffen.</li> <li>Sie lernen Architekturmuster für die geeignete Aufteilung von großen Softwareprojekten kennen und können diese für Softwareprojekte anwenden.</li> </ul>							
3	<b>Inhalte</b>							
	<ul style="list-style-type: none"> <li>Grundlagen, Motivation, Übersicht, Anforderungen.</li> <li>Entwurfsmuster, Architekturen, Frameworks.</li> <li>Microservices und monolithische Server-Architekturen.</li> <li>Planung und Entwicklung skalierbarer und sicherer Software-Komponenten.</li> <li>Responsive Single-Page Applikationen.</li> <li>Moderne Programmiersprachen (z.B. ECMAScript, Go, Rust).</li> <li>Frontend-Technologien (z.B. Angular, React, Vue).</li> <li>Backend-Technologien (z.B. Node.js, Django).</li> <li>Relationale und No-SQL Datenbanken (z.B. PostgreSQL, MongoDB).</li> </ul>							
4	<b>Teilnahmevoraussetzungen</b>							
	Formal: -, Inhaltlich: Kenntnisse der Objektorientierte Programmierung, Datenbanken, Software Engineering, Webbasierte Anwendungen, Cloud Computing							
5	<b>Prüfungsgestaltung</b>							
	Projektarbeit							
6	<b>Voraussetzung für die Vergabe von Credit Points</b>							
	Erfolgreicher Abschluss der Projektarbeit							
7	<b>Verwendung des Moduls</b> (in folgenden Studiengängen):							
	Bachelor Informatik							
8	<b>Modulbeauftragte/r</b>							
	Prof. Dr. Jörg Brunsmann							
9	<b>Sonstige Informationen</b>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodul aus Liste 2 „Anwendungen der Informatik“ Softwarequalität								Kürzel SQ
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
6.28	450h	15	5./6. Sem.	halbjährlich bei Nachfrage	SoSe/WS bei Nach- frage	1 Sem.	Wahl- pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>		<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>
	Vorlesung		2 SWS / 30h	60h	Wird in LV bekannt gegeben.		60	deutsch
	Praktikum		4 SWS / 60h	300h			15	deutsch
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>							
<p>Die Veranstaltung führt in Techniken und Methoden zur Softwarequalitätssicherung bei der Entwicklung und beim Betrieb von Softwaresystemen ein. Mit der erfolgreichen Teilnahme an der Veranstaltung wird ein Verständnis für Softwarequalität und die Bedeutung von systematischem Softwaretest erworben. Die Teilnehmer kennen den allgemeinen Testprozess sowie die Aufgaben der dabei beteiligten Rollen. Sie kennen verschiedene Teststufen und -arten und sind in der Lage, verschiedene statische und dynamische Prüftechniken und Werkzeuge bedarfsorientiert auszuwählen und einzusetzen. Die Teilnehmer sind in der Lage, unter Anwendung der erlernten Methoden Software von höherer Qualität zu entwickeln.</p> <p>Die Vorlesung dient der Vermittlung grundlegender theoretischer Kenntnisse und Fähigkeiten, wobei die praktische Anwendung im begleitenden Projekt erlernt und vertieft werden soll. Durch die teamorientierte Projektarbeit werden die Projektmanagement- und Selbst-Kompetenzen der Teilnehmer vertieft.</p> <p>Nach der erfolgreichen Teilnahme an diesem Modul ist optional eine Prüfung zum "Certified Tester - Foundation Level" nach ISTQB bei einem vom German Testing Board zertifizierten Prüfungsinstitut möglich.</p>								
3	<b>Inhalte</b>							
<p>Auswahl an Themen zur Wissensvermittlung:</p> <ul style="list-style-type: none"> <li>• Qualitätsaspekte von Softwaresystemen</li> <li>• Grundlagen des Softwaretestens, Testprinzipien, fundamentaler Testprozess</li> <li>• Testen im Software-Life-Cycle, Teststufen und -arten</li> <li>• Statische Testtechniken: Reviews, statische Analyse</li> <li>• Dynamische Testtechniken, Testdesign <ul style="list-style-type: none"> <li>– Spezifikationsbasiert (Black-Box): Äquivalenzklassen- und Grenzwerttest, Entscheidungstabellen, zustandsbasierter Test, weitere Black-Box-Entwurfsverfahren</li> <li>– Strukturbasiert (White-Box): Überdeckungen (C0, C1, C2, C3), Kontroll- und Datenfluss-Anomalien</li> <li>– Erfahrungsbasierter Test: Error Guessing, Exploratives Testen</li> </ul> </li> <li>• Testkonzept, Teststrategie, Testmanagement</li> <li>• Werkzeuge</li> <li>• Testautomatisierung</li> <li>• Testfallgenerierung</li> <li>• Wechselnde Inhalte der Projekte zu aktuellen Themen</li> </ul>								
4	<b>Teilnahmevoraussetzungen</b>							
keine								
5	<b>Prüfungsgestaltung</b>							
Projektarbeit								

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b> Bestandene Modulprüfung
<b>7</b>	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik
<b>8</b>	<b>Modulbeauftragte/r</b> Prof. Dr.-Ing. Carsten Gips
<b>9</b>	<b>Sonstige Informationen</b> <ul style="list-style-type: none"> <li>• Spillner, A., Linz, T.: "Basiswissen Softwaretest", dpunkt-Verlag, 2012</li> <li>• Kleuker, S.: "Qualitätssicherung durch Softwaretests", Springer Vieweg, 2013</li> <li>• Liggesmeyer, P.: "Software-Qualität", Springer Spektrum, 2009</li> <li>• Klaus Franz: "Handbuch zum Testen von Web- und Mobile-Apps", Springer Vieweg, 2014</li> <li>• Robert Martin: "Clean Code", Prentice Hall, 2008</li> <li>• Michael Feathers: "Working Effectively with Legacy Code", Prentice Hall, 2013</li> <li>• Roy Oshero: "The Art of Unit Testing", Manning, 2013</li> <li>• Gerard Meszaros: "xUnit Test Patterns", Addison Wesley, 2007</li> <li>• Kent Beck: "Test Driven Development", Addison-Wesley, 2002</li> <li>• Graham et al.: "Foundations of Software Testing", Cengage Learning, 2012</li> <li>• Myers, G.J.: "The Art of Software Testing", John Wiley, 2011</li> </ul>

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodul aus Liste 2 „Anwendungen der Informatik“ Webengineering								Kürzel WE
Nr.	Workload	Credit Points	Studien-semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
6.29	450h	15	5./6. Sem.	halbjährlich bei Nachfrage	SoSe/WS bei Nach-frage	1 Sem.	Wahl-pflicht	B.Sc.
<b>1</b>	<b>Lehrveranstaltungs-art</b>	<b>Kontaktzeit</b>	<b>Selbst-studium</b>	<b>Lehrformen (Lernformen)</b>	<b>gepl. Gruppengr.</b>	<b>Sprache</b>		
	Vorlesung	2 SWS / 30h	60h	Wird in LV bekannt gegeben.	60	deutsch		
	Praktikum	4 SWS / 60h	300h		15	deutsch		
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>							
	<p>Verständnis von aktuellen Konzepten, Methoden, Techniken, Werkzeugen und Erfahrungen zur ingenieurmäßigen Entwicklung von Web-Anwendungen sowie ihre praktische Anwendung in der eigenen Projektarbeit in Web-Entwicklerteams.</p> <ul style="list-style-type: none"> <li>• Bewerten von potentiellen Risiken von Web-Anwendungen</li> <li>• Befähigung, zukünftige Entwicklungen im Bereich des Web-Engineering zu verfolgen und zu beurteilen</li> </ul> <p>Die erworbenen Fähigkeiten tragen damit in besonderem Maße zur Ausprägung von spezifischen Analyse-, Design- und Realisierungskompetenzen und zur Erweiterung von spezifischen technologischen Kompetenzen bei. Teamorientierte Projektarbeit erhöht ferner Projektmanagement- und Selbst-Kompetenzen.</p>							
<b>3</b>	<b>Inhalte</b>							
	<p>Einführung in Web-Engineering (Motivation, Definition, Grundprinzipien)</p> <ul style="list-style-type: none"> <li>• Produktentwicklung</li> <li>• Requirements Engineering für Web-Anwendungen</li> <li>• Modellierung von Web-Anwendungen</li> <li>• Architektur von Web-Anwendungen</li> <li>• Testen von Web-Anwendungen</li> <li>• Web-Projektmanagement</li> <li>• Qualitätsaspekte (Usability, Performanz, Sicherheit)</li> <li>• javabasierte Web-Frameworks z.B. JSF mit Primefaces, Richfaces und JPA</li> <li>• Webanwendungen mit JavaScript und HTML5 <ul style="list-style-type: none"> <li>• Webpattern</li> <li>• Frameworks für Javascript z.B. Knockout JS, Angular JS, Node JS</li> </ul> </li> </ul>							
<b>4</b>	<b>Teilnahmevoraussetzungen</b>							
	<p><b>Formal:</b> -</p> <p><b>Inhaltlich:</b> Webbasierte Anwendungen, Objektorientierte Programmierung, Programmiermetdodik, Softwaretechnik, Datenbanken</p>							
<b>5</b>	<b>Prüfungsgestaltung</b>							
	Projektarbeit							
<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b>							
	Erfolgreicher Abschluss der Projektarbeit							
<b>7</b>	<b>Verwendung des Moduls (in folgenden Studiengängen):</b>							
	Bachelorstudiengang Informatik							
<b>8</b>	<b>Modulbeauftragte/r</b>							
	Prof. Dr.-Ing. Grit Behrens							
<b>9</b>	<b>Sonstige Informationen</b>							
	<ul style="list-style-type: none"> <li>• Kurz, Marinschek: „JavaSever Faces 2.2: Grundlagen und erweiterte Konzepte“, dpunkt 2013</li> <li>• Kappel, Pröll, Reich, Teschitzegger: Web-Engineering, dpunkt 2004</li> <li>• Backschat, Martin : „Enterprise JavaBeans und JPA“ Springer Spektrum 2016</li> <li>Tarasiewicz: „Angular JS - Framework“, dpunkt 2014</li> </ul>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodul aus Liste 2 „Anwendungen der Informatik“ Anwendungen der Künstlichen Intelligenz								Kürzel AKI
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
6.30	450h	15	5./6. Sem.	halbjährlich bei Nachfrage	SoSe/WS bei Nach- frage	1 Sem.	Wahl- pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>	
	Vorlesung Praktikum	2 SWS / 30h 4 SWS / 60h	60h 300h	V P		60 15	deutsch deutsch	
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Verständnis von aktuellen Konzepten, Methoden, Techniken, Werkzeugen und Erfahrungen zur ingenieurmäßigen Entwicklung von Anwendungen der Künstlichen Intelligenz sowie ihre praktische Anwendung in der eigenen Projektarbeit in Entwicklerteams. <ul style="list-style-type: none"> <li>• Bewerten von potentiellen Risiken von Anwendungen der Künstlichen Intelligenz</li> <li>• Befähigung, zukünftige Entwicklungen im Bereich der KI zu verfolgen und zu beurteilen</li> </ul> Die erworbenen Fähigkeiten tragen damit in besonderem Maße zur Ausprägung von spezifischen Analyse-, Design- und Realisierungskompetenzen und zur Erweiterung von spezifischen technologischen Kompetenzen bei. Teamorientierte Projektarbeit erhöht ferner Projektmanagement- und Selbst-Kompetenzen.							
3	<b>Inhalte</b> <ul style="list-style-type: none"> <li>• Anwendungsprojekte der KI werden in Teamarbeit erstellt</li> <li>• Anwendung von Methoden des Maschinellen Lernens (z.B. Neuronale Netze, Deep Learning, Support Vector Maschine, Entscheidungsbäume, Clusteringverfahren)</li> <li>• Sensoren (Video, Audio, Infrarotkamera, Elektrolumineszenzkamera, Wetterdaten, Raumluftparameter, Kennlimnienmessgeräte, Roboter, Kopter)</li> <li>• Bearbeiten von großen Datenmengen aus Forschungs- und Anwendungsprojekten der Dozenten in Teamarbeit</li> <li>• Merkmalsextraktion mit Elementen der Bildverarbeitung und Sprachverarbeitung</li> <li>• Anwendung von Bibliotheken modernen Tools für Datenanalyse und Machine Learning (z.B. Python, NumPy, Pandas, SciPy, Jupyter, IPython oder WEKA oder KNIME)</li> </ul> <p>Im Fokus steht weitestgehend die selbstständige Bearbeitung einer komplexeren Aufgabenstellung im Rahmen eines Entwicklungsprojekts im Team, welches auch in Kooperation mit Forschungs- und Entwicklungsabteilungen von Firmen bearbeitet werden kann. In der Regel bestehen die Projektgruppen aus 2-4 Studierenden, die sich frei zusammen finden, einen Projektleiter aus ihren Reihen wählen und nach mit den Dozenten vereinbarten Vorgehensmodellen entwickeln sollen. Der Dozent definiert die Zielsetzung und führt einen regelmäßigen Diskurs über den Fortgang des KI-Projekts</p>							
4	<b>Teilnahmevoraussetzungen</b> <b>Formal:</b> - <b>Inhaltlich:</b> Künstliche Intelligenz (Fach Liste 1- Methoden der Informatik) , Objektorientierte Programmierung, Datenbanken							
5	<b>Prüfungsgestaltung</b> Projektarbeit							
6	<b>Voraussetzung für die Vergabe von Credit Points</b> Erfolgreicher Abschluss der Projektarbeit							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

<b>7</b>	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik
<b>8</b>	<b>Modulbeauftragte/r</b> Prof. Dr.-Ing. Grit Behrens
<b>9</b>	<b>Sonstige Informationen</b> Ian H. Witten „Data Mining: Practical Maschine Learning Tools and Techniques“, ELSEWVIER 2017, ISBN 978-0128042915 Thomas Haslwanter “An Introduction to Statitics with Python“, Springer Nature 2016, ISBN 978-3-319-28316-6 Miroslav Kubat “An Introdruction to Maschine Learning“, Springer Nature 2017, ISBN 978-3-319-63912-3

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodul aus Liste 2: Game Programming								Kürzel GPR
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
6.31	450h	15	6. Sem.	halbjährlich bei Nachfrage	SoSe/WiSe bei Nach- frage	1 Sem.	Wahl- pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>	
	Seminaristischer Unterricht Praktikum / Seminar	2 SWS / 30h 4 SWS / 60h	60h 300h	Wird in LV bekannt gegeben.		60 15	deutsch deutsch	
2	<b>Lernergebnisse (learning outcomes)/ Kompetenzen</b>							
	<ul style="list-style-type: none"> <li>Die Studierenden kennen die Grundlagen der „einfachen“ Spieleprogrammierung, d.h. für klassische Computerspiele (z.B. Retro-Games oder Indy-Games, wie 2D-Jump'n Run, Maze, Simulation oder Board Games) mit einfacher Grafik ohne komplexe Game-Engines.</li> <li>Die Studierenden sind in der Lage, eigenständig ein Computerspiel nur mit Nutzung eines Hardware-Abstraction-Layers zu programmieren.</li> <li>In Rahmen eines Projektes haben die Studierenden ein solches Computerspiel z.B. unter Nutzung der C-API der libSDL2 in C/C++ implementiert.</li> </ul>							
3	<b>Inhalte</b>							
	<ul style="list-style-type: none"> <li>Grundlagen, Motivation, Übersicht, Anforderungen.</li> <li>Aufbau / Game Loop</li> <li>2D Grafik / Sprites / Animation</li> <li>Inputs und Events</li> <li>Sound</li> <li>Game AI</li> <li>Game programming patterns</li> </ul>							
4	<b>Teilnahmevoraussetzungen</b>							
	Formal: -, Inhaltlich: Kenntnisse der Programmierung, Software Engineering							
5	<b>Prüfungsgestaltung</b>							
	Projektarbeit							
6	<b>Voraussetzung für die Vergabe von Credit Points</b>							
	Bestandene Modulprüfung und ggf. Testat							
7	<b>Verwendung des Moduls (in folgenden Studiengängen):</b>							
	Bachelor Informatik							
8	<b>Modulbeauftragte/r</b>							
	Prof. Dr. Dr.-Ing. Matthias König							
9	<b>Sonstige Informationen</b>							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Wahlpflichtmodulaus Liste 2 „Anwendungen der Informatik“ Computer Vision								Kürzel CV
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
6.32	450h	15	5./6. Sem	halbjährlich bei Nachfrage	SoSe/WS bei Nachfrage	1 Sem	Wahl- pflicht	B.Sc.
1a	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>	
	Vorlesung	2 SWS / 30h	45h	Wird in LV bekannt gegeben		60	deutsch	
	Praktikum	4 SWS / 60h	315h			15	deutsch	
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b>							
	<p>Die Studierenden wenden gängige Methoden im Bereich Computer Vision an, und können diese mit geeigneten Softwarebibliotheken wie z.B. OpenCV auf Beispiele aus der Praxis übertragen. Darüber hinaus können sie ihre Ergebnisse mit geeigneten Metriken qualitativ und quantitativ bewerten.</p> <p>Im Rahmen einer Projektarbeit lernen die Studierenden außerdem praxisbezogene Projekte aus dem Bereich Computer Vision innerhalb einer vorgegebenen Frist selbständig und eigenverantwortlich in Teams durchzuführen. Sie sind in der Lage Lösungen zu finden, und in angemessener und verständlicher Form darzustellen. Durch eine teamorientierte Projektarbeit lernen die Studierenden Lösungswege in Gruppen zu finden und Aufgaben kooperativ zu bearbeiten.</p>							
<b>3</b>	<b>Inhalte</b>							
	<p>Computer Vision ist ein interdisziplinäres Arbeitsgebiet, in dem Methoden zur Analyse und Interpretation von Einzelbildern und Bildfolgen genutzt werden. Anwendungen finden sich beispielsweise in der Sicherheitstechnik, der Lokalisierung und Navigation von autonomen Fahrzeugen, oder der Rekonstruktion von Objekten. Die Anwendung aktueller Methoden für ausgewählte Themengebiete im Rahmen von Projekten ist wesentlicher Inhalt der Veranstaltung. Beispiele für mögliche Inhalte sind: Kamerakalibrierung, Merkmalsextraktion, Bildsegmentierung, Objekterkennung, Tracking.</p> <p>Ablauf der Projekte: In Absprache mit dem Dozenten wählen die Studierenden Themen aus und bearbeiten diese über ein Semester in Gruppen. Sie recherchieren den Stand der Technik und Wissenschaft, implementieren einen gewählten Lösungsansatz als funktionierende Software, evaluieren und dokumentieren ihrer Ergebnisse und stellen ihre Ausarbeitungen im Rahmen von Präsentationen vor.</p>							
<b>4</b>	<b>Teilnahmevoraussetzungen</b>							
	Keine							
<b>5</b>	<b>Prüfungsgestaltung</b>							
	<p>Klausur oder mündliche Prüfung oder schriftliche Hausarbeit oder Projektarbeit oder Performanzprüfung oder wissenschaftliches Poster oder Kurzpublikationsmanuskript oder Forschungsförderungsantrag oder Praktikums-, Exkursions- oder Tagesprotokoll oder Portfolio oder Lerntagebuch oder Parcourprüfung oder (gem. § 14 (4) RPO) eine Kombination aus verschiedenen Formen von Prüfungsleistungen. Prüfungen können in digitaler Form abgenommen werden.</p>							
<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b>							
	Testat und bestandene Modulprüfung							
<b>7</b>	<b>Verwendung des Moduls (in folgenden Studiengängen):</b>							
	Bachelorstudiengang Informatik							
<b>8</b>	<b>Modulbeauftragte/r</b>							
	Prof. Dr.-Ing. Jan Rexilius							
<b>9</b>	<b>Sonstige Informationen</b>							
	Literatur wird in der Veranstaltung bekannt gegeben.							



**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Praxisphase								Kürzel PRA
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
7.0	450h	18	7. Sem.	Jedes WS	WS	1 Sem.	Wahl- pflicht	B.Sc.
1	<b>Lehrveranstaltungs- art</b>		<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>		<b>gepl. Gruppengr.</b>	<b>Sprache</b>
	Tätigkeit bei individueller Praxisstelle		450h	-	Arbeitstätigkeit		1	deutsch
2	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Die Praxisphase bietet den Studierenden die Möglichkeit, ihre in mehreren Semestern an der Hochschule erworbenen fachlichen Fähigkeiten in der Praxis zu erproben und zusätzlich wichtige Kompetenzen im außerfachlichen Bereich zu erwerben. Es spielt daher im Rahmen einer praxisorientierten und arbeitsmarktgerechten Ausbildung sowie zur Persönlichkeitsbildung eine zentrale Rolle. Die Lernergebnisse umfassen: <ul style="list-style-type: none"> <li>• Orientierung im angestrebten Berufsfeld</li> <li>• Erwerb praktischer Kenntnisse und Kennenlernen berufstypischer Arbeitsweisen</li> <li>• Kennenlernen technischer und organisatorischer Zusammenhänge, die für das Berufsfeld typisch sind.</li> <li>• Beteiligung am Arbeitsprozess entsprechend dem Ausbildungsstand</li> <li>• Praktische Ausbildung an fest umrissenen, konkreten Projekten</li> </ul>							
3	<b>Inhalte</b> Praktische Tätigkeit mit deutlichem Informatik-Schwerpunkt, z.B. <ul style="list-style-type: none"> <li>• Durchführung von Anforderungsanalysen, Erstellung Softwareentwurf</li> <li>• Programmierung</li> <li>• Datenbankentwurf und -implementierung</li> <li>• Realisierung von Web-Anwendungen</li> <li>• Netzwerkplanung, Sicherheitsanalysen</li> <li>• Verarbeitung von Graphikdaten, Visualisierung</li> </ul> Rahmenbedingungen: <ul style="list-style-type: none"> <li>• Kürzere tägliche Arbeitszeiten als ein halber Tag sind nicht erlaubt.</li> <li>• Das gesamte Praktikum ist innerhalb von 9 Monaten abzuleisten.</li> <li>• Krankheits- und andere Ausfallzeiten zählen dabei nicht mit.</li> <li>• Das Berufspraktikum muss bei einer Praktikumsstelle absolviert werden.</li> <li>• Über das Berufspraktikum ist ein Bericht von 13 bis 20 Seiten anzufertigen.</li> <li>• Das Praktikum wird von einem hochschulseitigen Betreuer überwacht.</li> </ul>							
4	<b>Teilnahmevoraussetzungen</b> <b>Formal:</b> 110 cps (s. BPO) <b>Inhaltlich:</b>							
5	<b>Prüfungsgestaltung</b> Bewerteter Praxisbericht als Testat und Zeugnis der Ausbildungsstätte (siehe BPO)							
6	<b>Voraussetzung für die Vergabe von Credit Points</b> Bestandene Modulprüfung							
7	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
8	<b>Modulbeauftragte/r</b> Dozenten im Studiengang Informatik (Becking, Behrens, Brunsmann, George, Gips, Hoffmann, König, Kreienkamp, Müller, Thiel)							
9	<b>Sonstige Informationen</b> Arbeitsmaterialien und Literatur entsprechen der individuellen Aufgabenstellung							

**Modulhandbuch für den Bachelor-Studiengang Informatik  
des Fachbereichs Campus Minden**

Bachelorarbeit								Kürzel BA
Nr.	Workload	Credit Points	Studien- semester	Häufigkeit	Sem.	Dauer	Art	Q-Niveau
7.1	360h	12	7. Sem.	Jedes WS	WS	1 Sem.	Wahl- pflicht	B.Sc.
<b>1</b>	<b>Lehrveranstaltungs- art</b>	<b>Kontaktzeit</b>	<b>Selbst- studium</b>	<b>Lehrformen (Lernformen)</b>	<b>gepl. Gruppengr.</b>	<b>Sprache</b>		
	0,3 SWS Individuelle dozentengebundene Betreuung	10h	350h	Individuelle dozentengebundene Betreuung Selbständige Erstellung der Bachelorarbeit	1	deutsch		
<b>2</b>	<b>Lernergebnisse (learning outcomes) / Kompetenzen</b> Die selbständige Lösung eines praxisbezogenen Themas nach wissenschaftlichen Grundsätzen gehört zu den beruflichen Fähigkeiten eines Informatikers. Die systematische Bearbeitung und praxisbezogene Umsetzung einer Aufgabenstellung sowie die zusammenhängende Darstellung von Berichten und Publikationen dient der Kommunikation zwischen Fachleuten und stellt sicher, dass erworbene Kenntnisse und Fähigkeiten erhalten bleiben. Die Studierenden lernen, wie eine Aufgabe methodisch bearbeitet und in einem vorgegebenen Zeitrahmen mit einem klar strukturierten Ergebnis dargestellt wird, indem sie sich schnell in eine neue Aufgabenstellung einarbeiten und ihr Wissen in einem speziellen Gebiet selbstständig vertiefen. Die Studierenden lernen übliche Werkzeuge und Methoden zur Arbeitsunterstützung einzusetzen, wenden eine Reihe fachspezifischer Fähigkeiten, Fertigkeiten und Techniken an, um Aufgaben selbständig zu lösen, diese zu analysieren, zu bewerten und in einem Gesamtkontext darzustellen.							
<b>3</b>	<b>Inhalte</b> Mit der Bachelorarbeit soll unter Beweis gestellt werden, dass Studierende in der Lage sind, eine komplexe Fragestellung mit wissenschaftlichen Methoden innerhalb eines begrenzten Zeitraums zu lösen und das dabei erworbene theoretische und praktische Wissen nachvollziehbar zu dokumentieren. 1. Konkretisieren der Aufgabenstellung 2. Erstellung eines Zeitplans 3. Evaluation und Aufstellung der zu verwendenden Techniken und Methoden 4. Erstellung eines Software-Konzeptes 5. Implementierung und Dokumentation der Software-Lösung 6. Gesamtbetrachtung, Test und Bewertung der Lösung 7. Darstellung der Lösung in Form der Bachelorarbeit							
<b>4</b>	<b>Teilnahmevoraussetzungen</b> <b>Formal:</b> Bestandene Modulprüfungen gem. Studienverlaufsplan bis auf vier Modulprüfungen (siehe auch BPO). <b>Inhaltlich:</b> Kenntnisse in der Breite des studierten Faches							
<b>5</b>	<b>Prüfungsgestaltung</b> Von zwei Prüfern bewertete Bachelorarbeit							
<b>6</b>	<b>Voraussetzung für die Vergabe von Credit Points</b> Bestandene Modulprüfung							
<b>7</b>	<b>Verwendung des Moduls</b> (in folgenden Studiengängen): Bachelorstudiengang Informatik							
<b>8</b>	<b>Modulbeauftragte/r</b> Dozenten im Studiengang Informatik (Becking, Behrens, Brunsmann, George, Gips, Hoffmann, König, Kreienkamp, Müller, Thiel)							
<b>9</b>	<b>Sonstige Informationen</b> Arbeitsmaterialien und Literatur entsprechen der individuellen Aufgabenstellung							