

Module im Bereich: Informatik I - Pflicht

- [Algorithmen und Datenstrukturen](#)
- [Algorithms and Data Structures](#)
- [Database Concepts](#)
- [Datenbanken 1](#)
- [Einführung in die Informatik](#)
- [Introduction to Computer Science](#)
- [Modellierung](#)
- [Modeling](#)
- [Software Engineering & IT-Projektmanagement](#)

Algorithmen und Datenstrukturen

Sommer

(engl. Algorithms and Data Structures)

Modulnummer: FIN-INF-110361

Kürzel
AuD

CP
10

Semester
Sommer

Fachsem.
ab 2.

Dauer
1 Semester

Sprache
deutsch

Niveau
Bachelor

Link zum LSF:	LSF
Zugang beschränkt:	false
Verantwortung:	Christian Rössl
Dozent:in:	Christian Rössl
Lehrveranstaltungen:	In Deutsch / In German • Vorlesung <i>Algorithmen und Datenstrukturen</i> • Übung <i>Algorithmen und Datenstrukturen</i> • Tutorien <i>Algorithmen und Datenstrukturen</i> In Englisch / In English • Lecture <i>Algorithms and Data Structures</i> • Exercise <i>Algorithms and Data Structures</i> • Tutorial <i>Algorithms and Data Structures</i>
Verwendbarkeit:	- B.Sc. INF: Informatik I - Pflicht - B.Sc. INF: Kernmodule (SPO 2027) - B.Sc. CV: Informatik I - Pflicht - B.Sc. CV: Kernmodule (SPO 2027) - B.Sc. INGINF: Informatik I - Pflicht - B.Sc. INGINF: Kernmodule (SPO 2027) - B.Sc. WIF: Gestalten - Pflicht - B.Sc. WIF: Kernmodule (SPO 2027) - B.Sc. INF (bilingual): Informatik I - Pflicht - B.Sc. INF (bilingual): Kernmodule (SPO 2027)

Angestrebte Lernergebnisse:

Die Studierenden

- kennen und verstehen die wichtigsten grundlegenden Datenstrukturen und Algorithmen auf diesen
- können diese anwenden und beurteilen, welche für eine bestimmte Anwendung (besser) geeignet sind
- verstehen vorgegebene und erstellen eigene Implementierungen

Inhalt:

- Listen
- Bäume
- Balancierte Suchbäume
- Heaps
- Hashverfahren
- Graphen
- Dynamische Programmierung
- Suche in Texten
- Entwurf und Analyse von Algorithmen

Arbeitsaufwand:

106h Präsenz + 194h selbstständige Arbeit

Prüfungsvorleistungen:	Studien-/Prüfungsleistungen:	Lehrform / SWS:
Um an der Klausur teilnehmen zu können, sind Prüfungsvorleistungen erforderlich. Details werden in der ersten Vorlesung bekannt gegeben.	Klausur 120 Minuten	• Vorlesung (4 SWS) • Übung (2 SWS) • Tutorium (1 SWS)

Voraussetzungen nach Prüfungsordnung:	Empfohlene Voraussetzungen:
---------------------------------------	-----------------------------

keine	• Programmierkenntnisse in Java, insbesondere Objektorientierung und Abstrakte Datentypen • Grundlegendes Verständnis von Algorithmen, insbesondere Verständnis von Rekursion • Verständnis von asymptotischer Abschätzung des Aufwands von Algorithmen und deren formale Beschreibung (z.B. Groß-O-Notation) • Verständnis grundlegender Beweistechniken (z.B. vollständige Induktion)
-------	--

Medienformen:	Literatur:
	• Saake & Sattler: Algorithmen und Datenstrukturen • Goodrich, Tamassia & Goldwasser: Data Structures and Algorithms in Java • Sedgewick & Wayne: Algorithms • Cormen, Leiserson, Rivest, Stein: Introduction to Algorithms

Hinweise:

Die Abgabe des **ersten Übungsblatts** erfolgt ggf. schon in der ersten Semesterwoche, d.h. die Aufgaben können vorher bearbeitet werden! Dazu ist eine rechtzeitige Anmeldung auf

<https://aud.vc.cs.ovgu.de> erforderlich. Diese ist bereits vor Semesterbeginn möglich.

Algorithms and Data Structures

Summer

(in German: Algorithmen und Datenstrukturen)

Module-ID: FIN-INF-110487

Abbreviation	Credit Points	Semester	Term	Duration	Language	Level
DSA	10	Summer	starting at 2.	1 Semester	english	Bachelor

Link:	LSF
Limited accessibility:	false
Responsibility:	Christian Braune
Lecturer:	Christian Braune
Classes:	• Lecture <i>Algorithms and Data Structures</i> • Exercise <i>Algorithms and Data Structures</i> • Tutorial <i>Algorithms and Data Structures</i>
Applicability in curriculum:	- B.Sc. INF (bilingual): Informatik I - Pflicht - B.Sc. INF (bilingual): Kernmodule (SPO 2027)

Intended learning outcomes:

The students

- know and understand the most important basic data structures and algorithms on these
- are able to apply these and assess which are (better) suited to a specific application
- understand predefined and create their own implementations

Content:

- Lists
- Trees
- Balanced search trees
- Heaps
- Hash method
- Graphs
- Dynamic programming
- Search in texts
- Design and analysis of algorithms

Workload:

106h attendance + 194h independent work

Pre-examination requirements:	Type of examination:	Teaching method / lecture hours per week (SWS):
In order to be able to take part in the exam, preliminary work is required. Details will be announced in the first lecture.	Written exam 120 minutes	• Lecture (4 SWS) • Exercise (2 SWS) • Tutorial (1 SWS)

Prerequisites according to examination regulations:	Recommended prerequisites:
none	• Programming skills in Java, including Object Oriented Programming and Abstract Data Types • Basic understanding of algorithms, including understanding of recursion • Understanding of runtime/cost of algorithms and its formal representation (e.g., Big-O notation) • Understanding basic methods of proof (e.g., proof by induction)

Media:	Literature:
--------	-------------

- | | |
|--|---|
| | <ul style="list-style-type: none">• Saake & Sattler: Algorithmen und Datenstrukturen• Goodrich, Tamassia & Goldwasser: Data Structures and Algorithms in Java• Sedgewick & Wayne: Algorithms• Cormen, Leiserson, Rivest, Stein: Introduction to Algorithms |
|--|---|

Comments:

The **first assignment** may be due in the first week of the semester, which means you can work on the assignment in advance! This requires prior registration at <https://aud.vc.cs.ovgu.de>. Registration opens before the semester begins.

Database Concepts

Summer

(in German: Database Concepts)

Module-ID: FIN-INF-110489

Abbreviation	Credit Points	Semester	Term	Duration	Language	Level
DB1_en	5	Summer	3.	1 Semester	english	Bachelor

Link:	LSF
Limited accessibility:	false
Responsibility:	Gunter Saake
Lecturer:	Gunter Saake
Classes:	• Lecture <i>Database Concepts</i> • Exercise class <i>Database Concepts</i>
Applicability in curriculum:	- B.Sc. INF (bilingual): Informatik I - Pflicht - B.Sc. INF (bilingual): Kernmodule (SPO 2027)

Intended learning outcomes:

The students ...

- understand basic concepts of database systems
- are able to model relational databases
- are familiar with relational database languages
- can develop database applications

Content:

- Properties of Database Systems

- Architectures
- Conceptual Design of Relational Databases
- Relational Database Model
- Mapping of ER-Schemas to Relations
- Database Languages (Relational Algebra, SQL)
- Formal Design Criteria and Normalization
- Database Application Programming
- Further Database Concepts, e.g., Views, Triggers, Access Rights

Workload:

56 contact hours + 94 h self study

Pre-examination requirements:	Type of examination:	Teaching method / lecture hours per week (SWS):
Successful participation in the exercise	Written exam (120 min)	• 2 SWS Lecture • 2 SWS Exercise

Prerequisites according to examination regulations:	Recommended prerequisites:
Cannot be attended together with "Datenbanken 1"	

Media:	Literature:
	• Datenbanken - Konzepte und Sprachen. Gunter Saake, Kai-Uwe Sattler, Andreas Heuer, March 2013, ISBN 3-8266-9453-8, MitpVerlag; Edition 5 • Fundamentals of Database Systems. Ramez Elmasri, Shamkant B. Navathe, April 2010, ISBN 0-136-08620-9, Addison Wesley; Edition 6

Comments:

Datenbanken 1

Winter

(engl. Database Concepts)

Modulnummer: FIN-INF-100391

Kürzel
DB1

CP
5

Semester
Winter

Fachsem.
ab 1.

Dauer
1 Semester

Sprache
deutsch

Niveau
Bachelor

Link zum LSF:	LSF
Zugang beschränkt:	
Verantwortung:	Gunter Saake
Dozent:in:	Gunter Saake
Lehrveranstaltungen:	• Vorlesung <i>Datenbanken 1</i> • Übung <i>Datenbanken 1</i>
Verwendbarkeit:	- B.Sc. INF: Informatik I - Pflicht - B.Sc. INF: Kernmodule (SPO 2027) - B.Sc. CV: Informatik I - Pflicht - B.Sc. CV: Kernmodule (SPO 2027) - B.Sc. INGINF: Informatik I - Pflicht - B.Sc. INGINF: Kernmodule (SPO 2027) - B.Sc. WIF: Anwenden - Pflicht - B.Sc. WIF: Kernmodule (SPO 2027) - B.Sc. INF (bilingual): Informatik I - Pflicht - B.Sc. INF (bilingual): Kernmodule (SPO 2027)

Angestrebte Lernergebnisse:

Die Studierenden ...

- verstehen die Grundkonzepte von Datenbanksystemen
- können relationale Datenbanken entwerfen
- kennen relationale Datenbanksprachen
- können Datenbankanwendungen entwickeln

Inhalt:

- Eigenschaften von Datenbanksystemen
- Architekturen
- Konzeptueller Entwurf einer relationalen Datenbank
- Relationales Datenbankmodell
- Abbildung ER-Schema auf Relationen
- Datenbanksprachen (Relationenalgebra, SQL)
- Formale Entwurfskriterien und Normalisierungstheorie
- Anwendungsprogrammierung
- Weitere Datenbankkonzepte wie Sichten, Trigger, Rechtevergabe

Arbeitsaufwand:

56 h Präsenz + 94 h selbstständige Arbeit

Prüfungsvorleistungen:	Studien-/Prüfungsleistungen:	Lehrform / SWS:
Erfolgreicher Besuch der Übung	Schriftliche Prüfung (120 min)	• 2 SWS Vorlesung • 2 SWS Übung

Voraussetzungen nach Prüfungsordnung:	Empfohlene Voraussetzungen:
Kann nicht gleichzeitig mit dem Modul "Database Concepts" besucht werden	

Medienformen:	Literatur:
	Datenbanken - Konzepte und Sprachen. Gunter Saake, Kai-Uwe Sattler, Andreas Heuer. März 2018, ISBN 978-3-95845-777-5; Mitp-Verlag; Auflage: 6

Hinweise:

Einführung in die Informatik

Winter

(engl. Introduction to Computer Science)

Modulnummer: FIN-INF-110360

Kürzel	CP	Semester	Fachsem.	Dauer	Sprache	Niveau
Einf. INF	10	Winter	ab 1.	1 Semester	deutsch	Bachelor

Link zum LSF:	LSF
Zugang beschränkt:	false
Verantwortung:	Christian Rössl
Dozent:in:	Christian Rössl
Lehrveranstaltungen:	• Vorlesung <i>Einführung in die Informatik</i> • Übung <i>Einführung in die Informatik</i> • Tutorium <i>Einführung in die Informatik</i>
Verwendbarkeit:	- B.Sc. INF: Informatik I - Pflicht - B.Sc. INF: Kernmodule (SPO 2027) - B.Sc. CV: Informatik I - Pflicht - B.Sc. CV: Kernmodule (SPO 2027) - B.Sc. INGINF: Informatik I - Pflicht - B.Sc. INGINF: Kernmodule (SPO 2027) - B.Sc. WIF: Gestalten - Pflicht - B.Sc. WIF: Kernmodule (SPO 2027) - B.Sc. INF (bilingual): Informatik I - Pflicht - B.Sc. INF (bilingual): Kernmodule (SPO 2027)

Angestrebte Lernergebnisse:

Die Studierenden

- verstehen Grundkonzepte von Algorithmen, insbesondere auch Rekursion
- können Algorithmen und einfache Datenstrukturen in Java implementieren
- verstehen Grundlagen verschiedener Programmierparadigmen
- können Eigenschaften von Algorithmen und Datenstrukturen formal beschreiben (z.B. Korrektheit, Aufwand, ADT) und können grundlegende Beweistechniken anwenden

- wenden theoretische und praktische Inhalte in Übungen und Tutorien an

Inhalt:

- Algorithmische Grundkonzepte
- Grundkonzepte in Java
- Funktionen
- Objektorientierte Programmierung
- Programmierparadigmen
- Suchen und Sortieren
- Korrektheit von Algorithmen
- Aufwand von Algorithmen
- Abstrakte Datentypen und Einführung in grundlegende Datenstrukturen
- Berechenbarkeit und Entscheidbarkeit

Arbeitsaufwand:

- 98 h Präsenzzeit
- 202 h selbstständige Arbeit

Prüfungsvorleistungen:	Studien-/Prüfungsleistungen:	Lehrform / SWS:
Um an der Klausur teilnehmen zu können, sind Prüfungsvorleistungen erforderlich. Details werden in der ersten Vorlesung bekannt gegeben.	Klausur 120 Minuten	• Vorlesung (4 SWS) • Übung (2SWS) • Tutorium (1 SWS)

Voraussetzungen nach Prüfungsordnung:	Empfohlene Voraussetzungen:
keine	

Medienformen:	Literatur:
	Überwiegend: • Saake & Sattler: Algorithmen und Datenstrukturen Ergänzend: • Goodrich, Tamassia & Goldwasser: Data Structures and Algorithms in Java • Sedgewick & Wayne: Algorithms • Cormen, Leiserson, Rivest & Stein: Introduction to Algorithms

Hinweise:

Introduction to Computer Science

Winter

for BiBa

(in German: Einführung in die Informatik (für BiBa))

Module-ID: FIN-INF-110486

Abbreviation	Credit Points	Semester	Term	Duration	Language	Level
IntroCS	10	Winter	starting at 1.	1 Semester	english	Bachelor

Link:	LSF
Limited accessibility:	true
Responsibility:	PD Dr.-Ing. habil. Christian Krätzer
Lecturer:	PD Dr.-Ing. habil. Christian Krätzer
Classes:	• Lecture <i>Introduction to Computer Science</i> • Exercise <i>Introduction to Computer Science</i> • Tutorial <i>Introduction to Computer Science</i>
Applicability in curriculum:	- B.Sc. INF (bilingual): Informatik I - Pflicht - B.Sc. INF (bilingual): Kernmodule (SPO 2027)

Intended learning outcomes:

Students

- understand basic concepts of algorithms, in particular recursion
- can implement algorithms and simple data structures in Java
- understand the fundamentals of various programming paradigms

- can formally describe properties of algorithms and data structures (e.g. correctness, complexity, ADT) and apply basic methods of proof
- apply theoretical and practical content in exercises and tutorials

Content:

- Basic algorithmic concepts
- Basic concepts in Java
- Functions
- Object-oriented programming
- Programming paradigms
- Searching and sorting
- Correctness of algorithms
- Complexity of algorithms
- Abstract data types and introduction to basic data structures
- Computability and decidability

Workload:

- 98 h attendance + 202 h independent work

Pre-examination requirements:	Type of examination:	Teaching method / lecture hours per week (SWS):
In order to take the exam, preliminary exam requirements must be met. Details will be announced in the first lecture.	Written exam 120 minutes	• Lecture (4 SWS) • Exercise (2 SWS) • Tutorial (1 SWS)

Prerequisites according to examination regulations:	Recommended prerequisites:
none	

Media:	Literature:
	Basis: • Saake & Sattler: Algorithmen und Datenstrukturen Additional material: • Goodrich, Tamassia & Goldwasser: Data Structures and Algorithms in Java

Comments:

Die Teilnahme an dieser Lehrveranstaltung ist beschränkt auf die Studierenden des FIN Studienganges "Bilingualer Bachelor Informatik" (BiBa)

Modellierung

Winter

(engl. Modeling)

Modulnummer: FIN-INF-110127

Kürzel	CP	Semester	Fachsem.	Dauer	Sprache	Niveau
Mod	5	Winter	ab 1.	1 Semester	deutsch	Bachelor

Link zum LSF:	LSF
Zugang beschränkt:	
Verantwortung:	Dr.-Ing. Daniel Staegemann
Dozent:in:	Dr.-Ing. Daniel Staegemann
Lehrveranstaltungen:	• Vorlesung <i>Modellierung</i> • Übung <i>Modellierung</i>
Verwendbarkeit:	- B.Sc. INF: Informatik I - Pflicht - B.Sc. CV: Informatik I - Pflicht - B.Sc. INGINF: Informatik I - Pflicht - B.Sc. WIF: Gestalten - Pflicht - B.Sc. INF (bilingual): Informatik I - Pflicht

Angestrebte Lernergebnisse:

Studierende ...

- können Systeme, Daten und Prozesse mithilfe verschiedener Modellierungstechniken abbilden.
- können im Rahmen von Informations- und Kommunikationssystemen Lösungen gestalten, darstellen und kommunizieren.
- können komplexe, gestalterische Sachverhalte abstrahieren und analytisch denken.
- können Konzepte für Informatiklösungen und computergestützte Lösungen so entwerfen und darstellen, dass diese hierauf aufbauend implementiert werden können.
- können die von ihnen erworbenen Fähigkeiten der Abstraktion und Darstellung, besonders in Bezug auf die Abbildung von Prozessen, einerseits zur Erfüllung von Aufgaben an der Schnittstelle von Informatik, Wirtschaftsinformatik und Wirtschaftswissenschaften, aber

auch im Kontext von Fachgebieten außerhalb der Informatik einbringen.

- können die Darstellung ihrer Konzepte und Ideen an die Bedürfnisse unterschiedlicher Stakeholder anpassen und somit in zahlreichen Kontexten und über verschiedene Disziplinen hinweg einen wertvollen Beitrag zum Gelingen verschiedenartigster Projekte leisten.

Inhalt:

- Modellierungstheorie: Von der Diskurswelt zu formalisierten Informationsmodellen
- Prozesse, Workflows und Geschäftsprozesse
- Meta-Modelle, Referenzmodellierung
- Grundsätze ordnungsmäßiger Modellierung
- Grundlagen XML Fachkonzeptuelle
- Modellierung mit höheren Petri-Netzen, der Entity Relationship-Methode und der BPMN
- Objektorientierte Modellierung mit UML
- Umsetzung konkreter Aufgabenstellungen

Arbeitsaufwand:

- Präsenzzeiten: 28 SWS Vorlesung + 28 SWS Übung
- Selbstständiges Arbeiten: 54 h Vor- und Nachbereitung der Vorlesung + 40 h Entwicklung von Modellen für die Übung

Prüfungsvorleistungen:	Studien-/Prüfungsleistungen:	Lehrform / SWS:
	Benotete und unbenotete schriftliche Prüfung, 120 Min.	• Vorlesung (2 SWS) • Übung (2 SWS)

Voraussetzungen nach Prüfungsordnung:	Empfohlene Voraussetzungen:
keine	

Medienformen:	Literatur:
	Kecher, C. (2011): UML 2 – Das umfassende Handbuch. 4. Aufl. Reisig, W. (1998): Systementwurf mit Netzen. Berlin u. a.

Hinweise:

Modeling

Winter

(in German: Modellierung)

Module-ID: FIN-INF-110488

Abbreviation	Credit Points	Semester	Term	Duration	Language	Level
n	5	Winter	3.	1 Semester	english	Bachelor

Link:	LSF
Limited accessibility:	
Responsibility:	n.b.
Lecturer:	n.b.
Classes:	• Lecture <i>Modeling</i> • Exercise <i>Modeling</i>
Applicability in curriculum:	- B.Sc. INF (bilingual): Informatik I - Pflicht - B.Sc. INF (bilingual): Informatik Wahlpflicht (SPO 2027)

Intended learning outcomes:

Students ...

- are able to represent systems, data and processes using various modeling techniques.
- can design, present and communicate solutions within the framework of information and communication systems.
- can abstract complex, creative issues and think analytically.
- can design and present concepts for IT solutions and computer-aided solutions in such a way that they can be implemented on this basis.
- can apply the skills they have acquired in abstraction and representation, especially with regard to the mapping of processes, to fulfill tasks at the interface of computer science, business informatics and economics, but also in the context of subject areas outside of

computer science.

- can adapt the representation of their concepts and ideas to the needs of different stakeholders and thus make a valuable contribution to the success of a wide variety of projects in numerous contexts and across different disciplines.

Content:

- Modeling Theory: From the discourse world to formalized information models
- Processes, Workflows, and Business Processes
- Meta-models, Reference Modeling
- Principles of Proper Modeling
- Basics of XML
- Domain Conceptual Modeling with High-level Petri Nets, the Entity-Relationship Method, and BPMN
- Object-Oriented Modeling with UML
- Implementation of Concrete Tasks

Workload:

- Contact hours: 28 semester hours per week (SWS) of lectures + 28 SWS of exercises
- Self study: 54 hours of preparation and follow-up for the lectures + 40 hours of model development for the exercises

Pre-examination requirements:	Type of examination:	Teaching method / lecture hours per week (SWS):
	Graded und ungraded written examination, 120 Min.	• Lecture (2 SWS) • Exercise (2 SWS)

Prerequisites according to examination regulations:	Recommended prerequisites:
keine	

Media:	Literature:
	Kecher, C. (2011): UML 2 – Das umfassende Handbuch. 4. Aufl. Reisig, W. (1998): Systementwurf mit Netzen. Berlin u. a.

Comments:

Software Engineering & IT-Projektmanagement

Winter

(engl. Software Engineering & IT-Projectmanagement)

Modulnummer: FIN-INF-110476

Kürzel
SE&IT-PM

CP
5

Semester
Winter

Fachsem.
ab 3.

Dauer
1 Semester

Sprache
deutsch

Niveau
Bachelor

Link zum LSF:	LSF
Zugang beschränkt:	false
Verantwortung:	Dr. Thomas Wilde
Dozent:in:	Dr. Thomas Wilde
Lehrveranstaltungen:	• Vorlesung <i>Software Engineering & IT-Projektmanagement</i> • Übung <i>Software Engineering & IT-Projektmanagement</i>
Verwendbarkeit:	- B.Sc. INF: Informatik I - Pflicht - B.Sc. INF: Kernmodule (SPO 2027) - B.Sc. CV: Informatik I - Pflicht - B.Sc. CV: Kernmodule (SPO 2027) - B.Sc. INGINF: Informatik I - Pflicht - B.Sc. INGINF: Kernmodule (SPO 2027) - B.Sc. WIF: Gestalten und Anwenden - Wahlpflicht - B.Sc. WIF: Kernmodule (SPO 2027) - B.Sc. INF (bilingual): Informatik I - Pflicht - B.Sc. INF (bilingual): Kernmodule (SPO 2027)

Angestrebte Lernergebnisse:

Studierende, die den Kurs erfolgreich absolviert haben ...

- ... verstehen den Begriff "Software Engineering", seine Bedeutung und die Notwendigkeit strukturierter Softwareentwicklung.
- ... kennen verschiedene Prozessmodelle, deren Vor- und Nachteile sowie die Kernaktivitäten des Software Engineerings.
- ... wenden Testmethoden und Refactoring an, um Softwarequalität zu sichern und Fehler zu vermeiden.
- ... verstehen und nutzen agile Methoden wie Scrum und Kanban und vergleichen sie mit traditionellen Ansätzen.
- ... setzen Versionskontrolle, IDEs und CI/CD-Pipelines effektiv ein und verstehen Continuous Integration & Deployment.
- ... schreiben lesbaren, wartbaren und effizienten Code nach bewährten Prinzipien und Konventionen.
- ... schätzen Aufwand und Ressourcen ein, planen Projekte und überwachen deren Fortschritt.
- ... kennen Architekturkonzepte und Design-Patterns und wenden diese in Softwareprojekten an.
- ... arbeiten effektiv im Team, präsentieren Ideen klar und dokumentieren ihre Arbeit.

Inhalt:

- Software Engineering - Was ist das und wozu wird es gebraucht?
- Prozessmodelle: sequentielle & iterativ,
- Prozessaktivitäten: Spezifikation, Entwicklung, Validierung, Evolution,
- Qualitätssicherung: Testint & Refactoring,
- Agile Softwareentwicklung,
- Tools & Werkzeuge,
- Clean-Coding / Code-Conventions,
- Continuous Integration, Continuous Deployment, Continuous Delivery,
- IT-Projektmanagement: Schätzung & Prozessplanung,
- Softwarearchitektur,
- Design-Patterns,
- Praktische Beispiele.

Arbeitsaufwand:

- Präsenzzeit: 28h Vorlesung + 28h Übung
- 94h selbstständige Arbeit

Prüfungsvorleistungen:	Studien-/Prüfungsleistungen:	Lehrform / SWS:
-------------------------------	-------------------------------------	------------------------

• Um an der Prüfung teilnehmen zu können, sind Prüfungsvorleistungen erforderlich. Details werden in der ersten Vorlesung bekannt gegeben.	• Prüfung: schriftliche Klausur, 120 Minuten; • Schein (SPO bis 2022): Bearbeitung der Aufgaben & Teilnahme an den Übungen.	• Vorlesung (2 SWS) - Übung (2 SWS)
--	--	--

Voraussetzungen nach Prüfungsordnung:	Empfohlene Voraussetzungen:
keine	Erfolgreicher Abschluss von: • Einführung in die Informatik, • Algorithmen und Datenstrukturen.

Medienformen:	Literatur:
	• Ian Sommerville. <i>Software Engineering</i>, 10. Auflage. Pearson, 2016 • Erich Gamma, Richard Helm, Ralph E. Johnson, John Vlissides: <i>Design Patterns. Elements of Reusable Object-Oriented Software</i>, Addison Wesley, 1997 • Robert Martin: <i>Clean Code: A Handbook of Agile Software Craftsmanship</i>, Prentice Hall, 2008 Für alle Lehrbücher gilt, dass die Auflage und ggf. die Wahl zwischen englischem Original oder deutscher Übersetzung für uns irrelevant ist. Jede ältere Auflage ist in Ordnung.

Hinweise: