Modulbezeichnung	**Praktische Informatik 1**
Modulverantwortliche(r)	Prof. Dr. U. Bormann
Modulart	Pflicht/Wahl ☒ Wahlpflicht ☐
Spezialisierungsbereich	
Dauer des Moduls	1 Semester
Kreditpunkte	9 CP
Arbeitsaufwand	Berechnung des Workloads Präsenz 112 h Übungsbetrieb/Prüfungsvorbereitung 158 h Summe 270 h
Turnus des Moduls	angeboten in jedem WiSe
Voraussetzung für die Teilnahme	Keine ☒ Folgende
Lehr- und Lernformen	Seminar ☐ Vorlesung ☒ Tutorium ☒ Praktikum ☒ Projekt ☐
Lernziele	• Grundlegende Informatikkonzepte wiedergeben und erklären können. • Konzepte einer imperativen Programmiersprache kennen, verstehen und anwenden können. • Anschauliche Sachverhalte im Modell der Objektorientierung ausdrücken können. • Einfache Algorithmen entwickeln und in Java umsetzen können. • Einfache in Java realisierte Algorithmen systematisch testen können. • Probleme in Teilprobleme zerlegen und diese Strukturierung mit Mitteln von Java umsetzen und aussagekräftig dokumentieren können. • Formale Syntaxbeschreibungen verstehen können. • Eine einfache Entwicklungsumgebung nutzen können. • LaTeX zur Erstellung einfacher Dokumente nutzen können. • Versionsverwaltungssysteme einsetzen können. • In Gruppen Probleme analysieren und gemeinsam Lösungsstrategien entwickeln und präsentieren können. Die Vorlesungen Praktische Informatik 1 und 2 vermitteln essenzielles Grundwissen und Basisfähigkeiten, deren Beherrschung für nahezu jede vertiefte Beschäftigung mit Informatik – sowohl in der industriellen Anwendung, als auch in der Forschung – Voraussetzung ist.

Lerninhalte	1. Basiswissen: von Neumannsche Rechnerorganisation – Grundlagen der Rechnerarchitektur – Programm und Prozess – Programmiersprachen – Compiler, Assembler, Loader, Linker, Interpreter, Laufzeitumgebungen, Betriebssysteme – Grafische Benutzungsschnittstellen 2. Datenstrukturen: Information und ihre Repräsentation – Datentypen und Typanalyse – Elementare und zusammengesetzte Datentypen – rekursive Datentypen – Kanonische Operationen auf den eingeführten Datenstrukturen 3. Programmierparadigmen: (1) Imperative und funktionale Programmierung, (2) Objektorientierte (imperative) Programmierung, (3) Sequenzielle Programme versus nebenläufige Programme 4. Grundkomponenten imperativer Programmiersprachen: Schnittstellen und Ein-/Ausgabe, Variablen und Zuweisungen, Kontrollstrukturen, Blöcke, Funktionen, Rekursion 5. Syntax und Semantik imperativer Programmiersprachen: Syntax und Methoden der Syntax-Spezifikation, reguläre Ausdrücke, (erweiterte) Backus-Naur-Form (E)BNF 6. Prinzipien der objektorientierten Programmierung: Geheimnisprinzip – Methoden – Operationen – Objekte – Klassen – Botschaften – Ereignisverarbeitung – Attribute – Vererbung – Polymorphismus – Überladung – Generische Datentypen 7. Umsetzung der Punkte 2.-6. mit Java – Illustration anhand einfacher Algorithmen 8. Programmdokumentation und zugehörige Hilfswerkzeuge, z.B. JavaDoc 9. Testen von Programmen und zugehörige Hilfswerkzeuge, z.B. JUnit 10. Grundlagen der Netzwerkkommunikation: IP-Adressen, DNS, TCP, UDP 11. Grundkonzepte der Entwicklung graphischer Oberflächen Programmier-Praktikum: Programmentwicklung in Java – Realisierung einzelner, überschaubarer Programmieraufgaben Lehrveranstaltung(en): • 03-IBGP-PI1 Praktische Informatik 1: Imperative Programmierung und Objektorientierung
Prüfungsformen	KP; PL1: 70%, PL2: 30%; Portfolio, Klausur
Literatur	• David J. Barnes, Michael Kölling: Java lernen mit BlueJ - Objects first - Eine Einführung in Java. Aktuelle Auflage. Pearson Studium. Weitere Informationen (Beispielprogramme, Musterlösungen, im WWW verfügbare Literatur) sind auf der Web-Seite der Veranstaltung zu finden.