Praktische Informatik 1 Practical Computer Science 1		Modulnummer: INF-1		
Bachelor Pflicht Winf-Schwerpunkt-Pflicht Winf-Schwerpunkt-Wahlpflicht Winf-Wahl	Schwerpunkt Computational Finance E-Business IT-Management Logistik			
Anzahl der SWS $ \begin{array}{c c c c c c c c c c c c c c c c c c c $	Kreditpunkte: 9	Turnus angeboten in jedem WiSe		
Formale Voraussetzungen: -				
Inhaltliche Voraussetzungen: -				
Vorgesehenes Semester: 1. Semester				
Sprache: Deutsch				
	·	·		

Ziele:

- 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.

Inhalte:

- 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

Unterlagen (Skripte, Literatur, Programme usw.):

• 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.

Form der Prüfung:

KP; PL1: 70%, PL2: 30%; Portfolio, Klausur

Arbeitsaufwand	Präsenz Übungsbetrieb/Prüfungsvorbereitung	112 158	
Summe		270	
Lehrende:			Verantwortlich:

Lehrende: Verantwortlich:
Dr. T. Röfer, N.N. Prof. Dr. U. Bormann