



Universität Ulm

Bachelor of Science Wirtschaftsphysik (PO 2014)

Allgemeine Informatik I und II

Code 8242870002

ECTS-Punkte 12

Präsenzzeit 8

Unterrichtssprache Deutsch

Dauer 2 Semester

Turnus jedes Semester

Modulkoordinator Prof. Dr. Franz Schweiggert

Dozent(en) Dozenten der Informatik

Einordnung in die Studiengänge

- Mathematik BSc, Studienbeginn WiSe, Pflichtmodul, 1. und 2. Fachsemester
 - Mathematik BSc, Studienbeginn SoSe, Pflichtmodul, 2. und 3. Fachsemester
 - Wirtschaftsmathematik BSc, Studienbeginn WiSe, Pflichtmodul, 1. und 2. Fachsemester
 - Wirtschaftsmathematik BSc, Studienbeginn SoSe, Pflichtmodul, 2. und 3. Fachsemester
 - Mathematische Biometrie, Studienbeginn WiSe, Pflichtmodul, 1. und 2. Fachsemester
 - Physik B.Sc., 1.-3. Semester, Wahlpflicht bei Nebenfach Informatik
 - Wirtschaftsphysik B.Sc., 1.-3. Semester, Pflicht
 - Computational Science and Engineering BSc, Pflichtmodul, 1. und 2. Fachsemester
 - Chemieingenieurwesen BSc, Pflichtmodul, 1. und 2. Fachsemester
-

Vorkenntnisse keine

Lernergebnisse Die Studierenden sollen

- Grundlagen formaler Sprachen und ihre Definition kennen
 - mit Rechnern, Betriebssystemen, Dienstprogrammen und Werkzeugen praktisch umgehen können
-

- Einsicht und Intuition in der Konstruktion von Algorithmen anhand konkreter Beispiele besitzen
- Algorithmen anhand von Komplexitätsuntersuchungen beurteilen können
- in der Lage sein, in einer modernen Programmiersprache einfache Algorithmen systematisch zu entwickeln und in ein lauffähiges Programm umzusetzen
- komplexere Datenstrukturen wie etwa Bäume oder assoziative Arrays in Definition (Rekursion) und Anwendung (rekursive Algorithmen) kennen und verstehen
- die Prinzipien moderner Modellierungstechniken verstehen und auf der Ebene einfacher Aspekte anwenden können
- klassische wie auch moderne Programmierparadigmen (z.B. Rekursion, Abstrakte Datentypen, Vererbung, Polymorphie, Ausnahmenbehandlung) kennen und diese auch praktisch anwenden können

Inhalt

- Einführung in das verwendete Betriebssystem, Behandlung nützlicher Kommandos und Dienstprogramme sowie praktischer Umgang mit Dateien und Prozessen
- Formale Sprachen: Definition und Strukturierung
- Reguläre Ausdrücke, endliche Automaten
- Algorithmen und Komplexität
- Prinzipien der Systementwicklung und -strukturierung
- Typen von Programmiersprachen
- Standarddatentypen, einfache strukturierte Datentypen sowie Kontrollstrukturen der gewählten Programmiersprache
- Entwicklung von einfachen Algorithmen für Standardprobleme (z.B. Suchen, Sortieren)
- Strukturierung von Software im Großen
- Komplexe Datenstrukturen (z.B. Listen, Bäume) und Algorithmen darauf
- Moderne Programmiersprachenkonzepte wie Vererbung oder Polymorphie
- Aspekte der Verlässlichkeit (z.B. Ausnahmenbehandlung)
- Moderne Programmierertechniken (einfache Muster)

Literatur

- Knuth, D.: The Art of Computer Programming, Fundamental Algorithms; Addison-Wesley
- Wirth, N.: Algorithmen und Datenstrukturen; Teubner Verlag
- Lang, H.W.: Algorithmen und Datenstrukturen in Java; Oldenbourg
- Sedgewick, R.: Algorithmen in C; Addison Wesley
- Sedgewick, R.: Algorithmen in Java. Pearson Studium 2003
- Gamma, E. e.a., Entwurfsmuster, Elemente wiederverwertbarer objektorientierte Software, Addison-Wesley 2004

Lehr- und Lernformen

- Vorlesung (2x2 SWS)
- Übung mit Praktikum (2x2 SWS)

Arbeitsaufwand

Präsenzzeit: 112 h; Eigenstudium: Nacharbeitung (84 h), Übungsaufgaben (112h), Prüfung und Vorbereitung (52 h); **Summe: 360 Stunden**

Bewertungsmethode

Die Vergabe der Leistungspunkte erfolgt aufgrund des Bestehens je einer schriftlichen Modulteilprüfung in den beiden Lehrveranstaltungen Allgemeine Informatik I und II. Die Anmeldung zu jeder dieser Modulteilprüfungen setzt einen Leistungsnachweise voraus (Erreichen von 50 % der Punkte in den Übungsaufgaben).

Notenbildung Die Modulnote ergibt sich als leistungspunktgewichtetes Mittel aus den Ergebnissen der Modulteilprüfungen.

Grundlage für Numerik, Vertiefung in Informatik
