

# Programmietechnik 2

Unit 1: Übersicht, Ablauf

# Praktische Informatik

- Algorithmen und Datenstrukturen
  - Implementierung objektorientierter Konzepte
  - Algorithmische Komplexität
  - Sortieren und Suchen
  - Graphen
- Deklarative Programmierung

# Konzepte Objektorientierter Programmierung

- Programmieren im Großen
- Datenkapselung
- Komposition
- Nachrichtenaustausch
- Vererbung
- Polymorphie
- Verteilung von Verantwortlichkeiten

# Implementierung objektorientierter Konzepte

- Implementierungssprache: C
- Objektrepräsentation
  - Identität
  - Zustand
  - Verhalten
- Speicherverwaltung
  - Freispeicherverwaltung
  - Lebenszeit von Objekten
  - automatische Speicherverwaltung
- Spätes Binden
- dynamische Typinformation

# Programmiersprache Java

- Klassen, Objekte, Schnittstellen
- Deklaration von Datentypen
- Kontrollstrukturen
- Objekterzeugung und Lebensdauer von Objekten

# Algorithmische Komplexität

- Algorithmenbegriff
- Komplexitätsmaße
- asymptotische Komplexität
- Messen und Beweisen

# Sortieren

- Programmiersprache: Java
- Typunabhängige Algorithmen
- Klassifizierung von Sortieralgorithmen
- Selectionsort
- Insertionsort
- Bubblesort
- Quicksort
- Mergesort
- Heapsort

# Suchen

- Symboltabellen
- Sequentielle Suche
- Binäre Suche
- Bäume
  - binäre Suchbäume
  - AVL-Bäume
  - Rot-Schwarz-Bäume
  - B-Bäume
- Hashing
  - Hashfunktionen
  - statisches Hashing
  - offene Addressierung



# Graphen

- Repräsentation
- Suche in Graphen
- Gerichtete Graphen und Zusammenhang
- Minimaler Spannbaum
- Travelling Salesman Problem

# Deklarative Programmierung

- Prolog
  - Fakten
  - Regeln
  - Anfragen
- Prolog-Maschine
  - Unifikation
- Algorithmen in Prolog

# Literatur

- N.Wirth: *Algorithmen und Datenstrukturen*
- R.Sedgewick: *Algorithms in Java*
- Knuth: *The Art of Computer Programming, Vol.. III: Sorting and Searching*
- Cormen, Leiserson, Rivest: *Introduction to Algorithms*
- Clocksin / Mellish: *Programmieren in Prolog*
- Webseite:  
<http://www.dcl.hpi.uni-potsdam.de/teaching/pt2/>

# Übungen

- zweiwöchentliche Abgabe von Lösungen
  - alle zwei Wochen findet statt der Vorlesung ein Übungstermin statt
- Aufgaben können in Gruppen von 2 Studenten gelöst werden
- 50% der Punkte sind erforderlich zur Prüfungszulassung
- Abgabe der Lösung über die Webseite
- Realisierung der Aufgaben unter Linux