

# Einführung in die Informatik

## Brief Summary

---

Was Sie gelernt haben sollten

Wolfram Burgard  
Cyrill Stachniss

# Algorithmen

---

- Wie kann man Probleme beschreiben?
- Begriff des Algorithmus und Eigenschaften von Algorithmen
- Was kann man nicht berechnen und wieso nicht?
- Halteproblem

# Java

---

- Java als Möglichkeit Probleme zu modellieren und zu lösen
- Begriff des Modells
- Wie strukturiert man „Dinge“ in Java (Klassen, Objekte)
- Java Konzepte und Java Syntax
- Klassendefinitionen, Konstruktoren, Methoden, Variablen, etc.
- Erzeugen von Objekten, Referenzvariablen
- Overloading, Overriding, Vererbung, Zugriffsmethoden
- Klassen vs. Abstrakte Klassen vs. Interfaces

# Files und Streams

---

- Ein- und Ausgabe in Java
- Wozu dienen Streams, was sind deren Vorteile?
- Abstraktion durch Streams
- Flexibilität

# Kontrollstrukturen und Kollektionen

---

- Boolesche Ausdrücke
- If-then-else
- Schleifen (while, for)
- Vektoren und Arrays
- Listen und Bäume
- Wie realisiert man effiziente Operation (insert, remove, ...)
- Enumerations
- Wie erstellt man neue Container (z.B., Set Klasse)

# Aufwandsanalysen

---

- O-Notation: Idee, Definition, Vorteile vs. Laufzeitmessung, etc.
- Aufwandsanalysen durchführen
- Schleifen, verschachtelte Aufrufe, etc.
- Grundlegende Komplexitätsklassen
- Suchen und Sortieren

# Rekursion

---

- Idee der Rekursion
- Rekursion vs. Iteration (Schleifen), Vor- und Nachteile
- Von der Iteration zur Rekursion
- Activation Records
- Endrekursion
- Umsetzung von rekursiven Funktionen

# Programmierpraxis

---

- Praktische Umsetzung der gelernten Konzepte
- Klassen definieren, Konstruktoren und Methoden realisieren
- Einfache Vererbung
- Zugriffsmethoden realisieren
- Durchläufe durch Container
- Methoden umsetzen
- Programmcode erweitern
- Implementierung einfachen rekursiven Funktionen
- Fehler in gegebenem Programmcode finden
- ...
- **Die Übungen sind die beste Klausurvorbereitung**