

Inhalt dieser Präsentationskopie

Anklickbare Hyperlinks: (Alternativ: *Bookmarks* einblenden oder von nachfolgender Slide nutzen → Index mit über 100 Hyperlinks)

- | | |
|--|----------------------------------|
| 1. Ein Algorithmus und seine Implementierung in Java | 9. Suchbäume |
| 2. Java: Elementare Aspekte | 10. Backtracking |
| 3. Klassen und Referenzen | 11. Spielbäume |
| 4. Syntaxanalyse und Compiler | 12. Rekursives Problemlösen |
| 5. Java: Pakete | 13. Komplexität von Algorithmen |
| 6. Objektorientierung | 14. Modellierung und Simulation |
| 7. Java: Weitere Sprachelemente | 15. Heaps |
| 8. Binärsuche auf Arrays | 16. Parallele Prozesse / Threads |

Relevant für die Prüfung ist der gesamte Inhalt der Vorlesung und Übungen, nicht nur diese Präsentationskopie – sie stellt auch keinen Ersatz für das genannte Lehrbuch dar.

Seiten mit Zusatz- und Hintergrundmaterial (wie z.B. historische Anmerkungen) sind am unteren Rand violett markiert:

(Diese können durch „Klick“ übersprungen werden)



Inhaltsverzeichnis

1. EIN ALGORITHMUS IN JAVA

Zum Begriff „Algorithmus“

Al-Chwarizmi

Altägyptische Multiplikation

Leibniz (Dualsystem, Rechenmaschine, Mechanisierung des Intellekts)

Überlauf als Fehler

Korrektheit, Verifikation

Induktion, Russell, Frege

(Schleifen)invarianten

Hoare-Kalkül

Proof-Carrying Code, Validierung

Aufwand, Effizienz

Kryptographie, Sicherheit

2. ELEMENTARES JAVA

Arrays

Typkonversion, Hüllenklassen

Ein- / Ausgabe, Strings

3. KLASSEN UND REFERENZEN

Class „Datum“

Getter-, Setter-Methoden

this

static (Variablen)

static (Klassenmethoden)

Osterdatum

Komputistik

4. SYNTAXANALYSE UND COMPILER

Bäume, Wurzelbäume

Beispiele für Bäume

Baumdarstellungen

Zeichen, Bedeutung

Binärbäume

Syntaxanalyse

Rekursiver Abstieg

Operatorbäume, inorder, postorder

Euler, Königsb. Brückenproblem

Postfix

Rechenschemata

Adam Ries, Abakus

Codegenerierung, Automat

Stackmaschine (Codeerzeugung, Interpreter)

ETH 1950: Z4, ERMETH, H. Rutishauser, A. Speiser

Elektronenröhren

ERMETH

Wer erfand das Programmieren?

Babbage, Menabrea, Ada

Bytecode, Java-VM

5. PAKETE IN JAVA

Bruchrechnen

Euklidischer Algorithmus, ggT

6. OBJEKTORIENTIERUNG

Konzepthierarchie, Kategorien, Instanzen

Vererbung, Polymorphie

7. WEITERE JAVA-SPRACHELEMENTE

Interfaces

Exceptions

ArrayList, Generics

8. BINÄRSUCHE AUF ARRAYS

Lineare Interpolationssuche

Lukas 11,9-1

9. BINÄRE SUCHBÄUME

Referenzen: Parameterübergabe immer „by value“!

Inorder-Traversierung

Binary tree sort

10. BACKTRACKING

Labyrinth, edge-matching puzzle

n-Damen-Problem

11. SPIELBÄUME, COMPUTERSPIELE, SPIELTHEORIE

Tic-Tac (Davies, Kates, DEUCE)

NIM, Nimrod

Computerschach, Schachtürke

Schach: Babbage, Torres Quevedo

Norbert Wiener, Kybernetik

1950er-Jahre: Machine Age, Bewusstsein der Maschinen

Kybernetik im Sozialismus

Heinz Zemanek

Automatisches Schachspiel:

Alan Turing, Claude Shannon, Dietrich Prinz

Schachprogramme, Schachcomputer Go

Automatisierung: Begriffsrelevanz in den 1950er- / 1960er-Jahren

Spieltheorie

John von Neumann und

Oskar Morgenstern

Spielstrategien, Minimax

George Boole, Claude Shannon

Spielbaum-Auswertung

Alpha-Beta

Reversi

12. REKURSIVES PROBLEMLÖSEN

Hilbert-Kurve, Turtle-Grafik

Divide et impera

Türme von Hanoi

Mergesort, Quicksort

Ors eirt, beis eilp!

13. KOMPLEXITÄT VON ALGORITHMEN

O-Notation

14. MODELLIERUNG UND SIMULATION

Modelle, Modellierung

Simulationsanwendungen (Beispiele)

Wettermodelle: Simulation, Prognose

Richardson's Forecast Factory

Menschliche Computer

Wetterprognose ohne Computer

Wetterprognose mit Computer

ENIAC

Supercomputer für numerische Wetterprognosen

Satellitenwetter

Chaos und Ensembles

Heutige Wetter- und Klimamodelle

Zeitgesteuerte Simulation

Weizen-Mäuse-Katzen-Beispiel

Grenzen des Wachstums (1972)

Ereignisgesteuerte Simulation

Priority Queue

15. HEAPS

Heapsort

Vergleich von Sortierverfahren

16. PARALLELE PROZESSE UND THREADS

Prozess- / Threadzustände

Thread-Erzeugung, start

Moore's Law

Thread-Steuerung (yield, wait, sleep, suspend, resume, stop)

join, Rendezvous-Synchronisation

Thread-Scheduling, Prioritäten

Parallele Threads auf dem Mars

Race condition, lost update

Atomarität, Inkonsistenz, kritische Abschnitte

E.W. Dijkstra

Dijkstra-Algorithmus (kürzeste Wege)

synchronized

Deadlock

RESÜMEE DER VORLESUNG

Abstraktion

Bonus-Slides: Geschichte, Kontext

~~Wozu wurden Computer erfunden?~~

~~Astronomie und Schusstabellen~~

Prüfungsvorbereitung

Nachdenkliches zum Schluss

Wer sind wir?

Fachgebiet „**Verteilte Systeme**“
im Departement Informatik



**Friedemann
Mattern**

Lukas Burkhalter
Ansprechperson für
organisatorische Aspekte
(z.B. Übungsbetrieb)



Mit was beschäftigen wir uns sonst?



Interaction

Smart Environment

Communication

Context Awareness

Web Technologies

Internet of Things

Cyber-Physical Systems

Service Discovery

Sensor Networks

Privacy

Ubiquitous Computing

Security

Augmented Reality

Social Impact

Smart Energy

Web of Things

Distributed Systems

Mehr zu uns:

www.vs.inf.ethz.ch

Und die Tutoren

Vincent Becker, Lukas Burkhalter, Mihai Bâce, Jing Yang, Alexander Viand, Julia Gygax, Pascal Schärli, Carina Fuss



ZUR BEACHTUNG

(das Kleingedruckte)

Diese Vorlesung ist Teil der „[Critical-Thinking-Initiative](#)“ der ETH Zürich, die damit eine Kultur des kritischen Denkens und eigenverantwortlichen Handelns fördern will.

In diesem Sinne wird zum historischen und gesellschaftlichen Kontext der angesprochenen fachlichen Themen Hintergrundmaterial (im Format von Slides mit einer Fussleiste in hellem Violett) bereitgestellt, das nicht in der Vorlesung selbst gezeigt wird (und keinen Prüfungsstoff darstellt). Diese „Bonusslides“ enthalten teilweise Zitate oder längere Passagen historischer Texte, ebenso Bildmaterial vergangener Jahrzehnte und Jahrhunderte. Sprachdiktion, Vokabular, Wortbedeutung, bildliche Darstellungen und teilweise auch die damit transportierten gesellschaftliche Normen und Werte entsprechen dabei nicht immer den heutigen Gewohnheiten und mögen daher aus jetziger Sicht manchmal sonderbar, einseitig, anstössig, unfair, oder gar diskriminierend wirken.

Einige Beispiele sollen dies verdeutlichen: Geldleiher waren im Mittelalter Juden (da den Christen Geldgeschäften mit Zins aus religiösen Gründen verboten war). Einen Geldleiher sprach man nicht mit „Sehr geehrter Herr Bankier“ an, sondern eher mit „Ich bitt` euch Jud, leicht mir zuo Hand Bargelt...“. Die 17-jährigen Sprösslinge reicher Basler Handelsleute, die anderes im Kopf hatten, als in einer Schule zu sitzen, bildete der gleichaltrige (und später sehr berühmte Renaissance-Maler) Hans Holbein d. J. explizit in ziemlich „unvorteilhafter“ Körperhaltung und in stutzerhafter Kleidung mit weiten Hemden und knappen Wämsen ab; das Original hängt heute im Kunstmuseum Basel. Unverheiratete Frauen wurden damals selbstverständlich als „Jungfrauen“ bezeichnet, und noch im letzten Jahrhundert gab es sogenannte „Rechenmädchen“ (analog wie wir heute noch immer von „Kindermädchen“ und eher selten von „Kinderfrau“ sprechen), die in den Büros die (mechanischen) Rechenmaschinen bedienten. Eine Grosstante von mir war als Rechenfräulein in einer Bank tätig – die ganze Familie war auf Tante Emmy stolz, denn sie hatte ja eine sehr anspruchsvollere Tätigkeit! In den USA sprach man von den „girls“ – „office girls“ waren allgegenwärtig; sie nannten sich auch untereinander so. Zur Bedienmannschaft des ENIAC-Computers gehörten dann entsprechend die „ENIAC girls“ etc.

ZUR BEACHTUNG (2)

Geschichte ist oft nicht so, wie wir gerne hätten, dass sie gewesen sei. Wir sollten sie aber dennoch anerkennen. Die historischen Texte und Zitate stellen insofern ein Zeitzeugnis dar – deren Werte man sich aber natürlich in der heutigen Zeit nicht zu eigen macht! Analoges gilt auch für die geschlechtsspezifischen Rollen und Berufe; diese waren früher weit stärker ausgeprägt als heute, was vor allem bei seinerzeitigen Abbildungen auffällt, die oft stereotypisch wirken. Zwar ist insbesondere die Werbung auch heute noch stark stereotypisch geprägt, inzwischen überkommene Rollenbilder in Abbildungen vergangener Zeiten fallen uns aber besonders deutlich auf und sehen oft befremdlich aus – heute würde man mit einem in Szene gesetzten Familienideal der 1950er-Jahre Kundinnen und Kunden eher abschrecken als gewinnen.

Ein Anliegen der Vorlesung ist es auch zu zeigen, dass neue Ideen oft mit Personen verbunden waren, die auf ihre Art vielleicht genial waren, aber ansonsten Menschen wie du und ich waren, in ihrer eigenen Zeit lebten und es oft nicht einfach hatten, ihren ungewohnten Ideen zu Anerkennung zu verhelfen. Dafür werden oft Zitate und Texte aus Biographien bemüht, die manchmal auch Passagen mit scheinbar Nebensächlichem enthalten. Auch Biographien oder zeitgemässe schriftliche Dokumente wie Briefe sind nicht „neutral“ oder gar „politisch korrekt“, und insbesondere Biographien und Nachrufe neigen zu einer Heroisierung der betreffenden Person („...wollte unser Volk befähigen, seinen Platz an der Sonne zu behaupten“).

Bei all dem beachte man generell, dass die Texte ein Produkt ihrer Zeit sind. Sie stehen im Allgemeinen für sich und werden nicht moralisierend kommentiert oder historisch „geframed“. Historische Texte, aber auch zeitgenössische aktuelle Beispiele von Webseiten und Zeitungsmeldungen, sind aber natürlich keineswegs immer als „Vorbilder“ gemeint, sondern stellen oft sogar Negativbeispiele dar: „Seht her, wohin das nun geführt hat, und schaut mal, wie ignorant oder abfällig sich Zeitungen heute über etwas äussern.“ Das können Charakterisierungen von „weather girls“ im Privatfernsehen sein (ein unerwartetes Nebenergebnis des alten Traums der Meteorologen, das Wetter in Echtzeit berechenbar zu machen) oder abfällige Bemerkungen im „Spiegel“ zum Kybernetik-Erfinder Norbert Wiener oder zu Gottfried Wilhelm Leibniz, immerhin der „Schutzpatron der Informatik“!

ZUR BEACHTUNG (3)

Insofern bleibt es Ihnen überlassen, geschätzte Studentinnen und Studenten, die Texte „richtig“ zu interpretieren und in „gut“ und „schlecht“ (wenn denn überhaupt die Welt und ihre Moral so einfach dichotom ist!) einzuordnen und zu verstehen, ob sie einen Sachverhalt beispielhaft verdeutlichen oder eher im Sinne eines Negativbeispiels schildern. Das macht mitunter Mühe – es erfordert „ein hinreichendes Mass an Kontextverständnis zur Einordnung des Wissens“, um noch einmal die Critical-Thinking-Initiative zu zitieren. Und weiter heisst es dort: „Ausgehend von diesem Ansatz geht es unter dem Begriff „critical thinking“ um die Förderung der Fähigkeit von Studierenden, verschiedene Positionen, Sichtweisen und Anspruchshaltungen zu unterscheiden.“ Und wozu das Ganze? Auch dazu findet sich natürlich eine Antwort bei der Initiative: „Ziel der ETH ist es, ihre Studierenden zu kritischen und unabhängigen denkenden Persönlichkeiten auszubilden.“ So gesehen könnte man (und wir bleiben mit Absicht im Konjunktiv) fast wörtlich einen Disclaimer übernehmen, den eine hier ungenannt bleibende Institution verwendet – wenn wir als ETH das denn nötig hätten (und plagieren dürften):

„Es wird festgestellt, dass die auf diesen Seiten (in Wort und Bild) abgebildeten Inhalte weder in irgendeiner Form propagandistischen Zwecken dienen, noch zum Zwecke der Herabwürdigung der Menschenwürde gezeigt werden. Die Inhalte dienen ausschliesslich historischen, sozial- oder kulturwissenschaftlichen Bildungszwecken. Ihre Veröffentlichung erfolgt in der Absicht, Wissen zur Anregung der intellektuellen Selbstständigkeit und Verantwortungsbereitschaft des Staatsbürgers zu vermitteln und damit der Förderung seiner Mündigkeit zu dienen.“

So traurig-ernst offiziell ist es aber hier gar nicht gemeint. Geschichte und Kontextwissen machen (fast allen) Spass – überzeugen Sie sich selbst!

-- Friedemann Mattern.

...total Schrecklich! Wieso muss es alle diese Bonuslides enthalten? Niemand lies dem. wir sind alle Studente. Niemand hat eine extra Interesse auf dinge wo nicht Prüfungsrelevant sind, und fast niemand redet Französisch. [Vorlesungsfeedback]

Themen der Vorlesung

Programmieren in Java

1. Ein Algorithmus und seine Implementierung in Java
2. Java: Elementare Aspekte
3. Klassen und Referenzen
4. Syntaxanalyse und Compiler
5. Java: Pakete
6. Objektorientierung
7. Java: Weitere Sprachelemente
8. Binärsuche auf Arrays

Allg. Informatik-Grundkonzepte

Algorithmen und Datenstrukturen

9. Suchbäume
10. Backtracking
11. Spielbäume
12. Rekursives Problemlösen
13. Komplexität von Algorithmen
14. Modellierung und Simulation
15. Heaps
16. Parallele Prozesse / Threads

Anwendungsaspekte

Um was geht es in der Vorlesung?

(Zitat aus dem Vorlesungsverzeichnis)

- **Algorithmen** und **Datenstrukturen**
- Fortgeschrittenes **Programmieren**
 - Java, Objektorientierung
- Prinzipielle **Grundlagen**
 - Abstraktion, Modellbildung, Formalisierung, Korrektheit,...
- Teilaspekte aus **Anwendungen**
 - Simulation, Multitasking
 - „praktische Informatik“

Kurzbeschreibung (Diploma Supplement)

Die Vorlesung vermittelt die gebräuchlichsten Problemlösungsverfahren, Algorithmen und Datenstrukturen. Themen sind u.a.: Divide & Conquer-Prinzip, Rekursion, Sortieralgorithmen, Backtracking, Suche in Spielbäumen, Datenstrukturen (Listen, Stacks, Binärbäume etc.) zeitdiskrete Simulation, Nebenläufigkeit, Komplexität, Verifikation. Bei den Übungen wird die Programmiersprache Java verwendet.

Lernziel

Einführung in die Methoden der Informatik sowie Vermittlung von Grundlagen zur selbständigen Bewältigung von anspruchsvolleren Übungen und Studienarbeiten mit Informatikkomponente im nachfolgenden Bachelor- und Masterstudium.

Inhalt

Die Vorlesung vermittelt die gebräuchlichsten Problemlösungsverfahren, Algorithmen und Datenstrukturen. Der Stoff umfasst auch grundlegende Konzepte und Mechanismen der Programmstrukturierung. Darüber hinaus wird generell das Denken in formalen Systemen, die Notwendigkeit zur Abstraktion, sowie die Bedeutung geeigneter Modellbildungen für die Informatik motiviert. Der Schwerpunkt der Vorlesung liegt auf der praktischen Informatik; konkrete Themen sind u.a.: Komplexität und Korrektheit von Algorithmen, Divide and Conquer-Prinzip, Rekursion, Sortieralgorithmen, Backtracking, Suche in Spielbäumen, Datenstrukturen (Listen, Stacks, binäre Bäume etc.), zeitdiskrete Simulation, Nebenläufigkeit, Verifikation. Bei den praktischen Übungen wird die Programmiersprache Java verwendet, dabei werden auch Aspekte wie Modularisierung, Abstraktion und Objektkapselung behandelt. Gelegentlich werden auch kurze Hinweise zum geschichtlichen Kontext der jeweiligen Konzepte gegeben. In den Übungen wird u.a. in Gruppen ein Spielprogramm für „Reversi“ programmiert, am Ende des Semesters findet dazu ein Turnier statt.

Um was geht es NICHT in der Vorlesung?

- Sie ist **KEIN** umfassender **Java-Programmierkurs!**
 - Programmieren mit Java eher Thema in praktischen **Übungsaufgaben** und der **Tutorien**
- Vorlesungsthema sind aber einige relevante Programmier**konzepte** am Beispiel von Java
 - Nötig auch für die Prüfungsklausur
- Bezüglich Java sind **Vorlesung / Tutorien** **komplementär**
 - **Programmierpraxis** mittels Übungsaufgaben trainieren (Klavierspielen lernt man auch nicht durch Zuhören eines Pianisten!)
 - Zu Java wird hier vorausgesetzt, dass man **C++ beherrscht** (auf dem Niveau von „Informatik I“: darauf bauen wir auf)



Programmieren vs. Konzepte

*Theorie ohne Praxis ist leer,
Praxis ohne Theorie ist blind.*
-- Immanuel Kant

■ Programmieren

- Wichtig, da Voraussetzung für viele Berufe
- Faktenorientiert
- Erfordert Training
- Muss selbst praktiziert werden
- Üben und lernen aus Fehlern
- Kostet relativ viel Zeit
- → **Tutorien**, Übungsaufgaben





Vorlesung und Übungen sind **komplementär**, nicht disjunkt

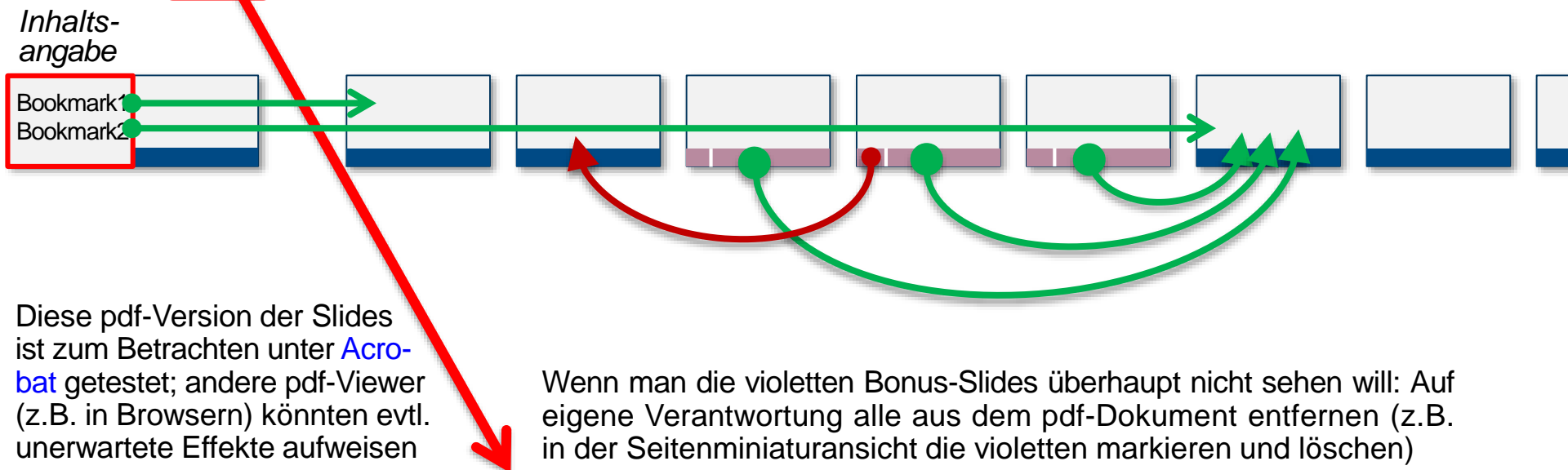
■ Konzepte

- Erkenntnisorientiert
- Verallgemeinerbares Wissen
- Länger relevante und bedeutsamere Kenntnisse
- Auch das praktische Programmieren fundiert auf Konzepten
- → **Vorlesung** (Abklärung und Festigung → Tutorien, Übungsaufgaben)

Organisatorisches

*Theorie ohne Praxis ist steril,
Praxis ohne Theorie unfruchtbar.*
-- Gregor Snelting, KIT

- **Vorlesungswebseite** für weitere Informationen
 - www.vs.inf.ethz.ch/edu/I2/ (bzw. auch via <http://lec.inf.ethz.ch>)
- **Folienkopien** zum Herunterladen dort
 - Diverse Formate (auch zum Ausdrucken) – update Semesterende
 - Empfehlung: Einmalig herunterladen und **lokal speichern** (~ 115 MB)
 - 50% Zusatz- / Hintergrundmaterial: Fussleiste in hellem Violett: 
 - **Klick** darauf → Sprung zur nächsten „echten“ Vorlesungsfolie: 



Organisatorisches (2)

- **Vorlesung:** 10:15 – 11:00, 11:10 – 11:55 (ETH-Reglement!)
- **Übungsbetrieb** beginnt nächste Woche!
 - Baldmöglichst **Anmeldung via CodeExpert** für Übungsgruppen!
 - Anmelde-link: <https://expert.ethz.ch/enroll/AS19/ifee2>
 - Gleiche Gruppe während des Semesters (→ zusammengehöriges **Team**)
 - **Morgen 13:15 – 15:00, HG E 7: Einführung Java / Infos Übungsbetrieb**
- **Aufgabenblatt** jede Woche via CodeExpert
- **Prüfung:** Sessionsprüfung schriftlich als Klausur
 - Prüfungstoff: Inhalt der Vorlesung + Übungsaufgaben / -stunden
 - **Bonus:** ¼-Note (max.) bei Lösen speziell gekennzeichnete Bonusübungen → Nur selbständig verfasste Lösungen abgeben (z.T. automatisierte Plagiatprüfung etc. → ggf. disziplinarische Massnahmen)
- **Kompliziertere Fragen** zur Vorlesung: Am besten in den Pausen
 - Oder in den Übungen an den Tutor oder per E-Mail an uns

Materialien und ergänzende Literatur

1) Folienkopien

- Zum Herunterladen als pdf auf der Vorlesungswebseite



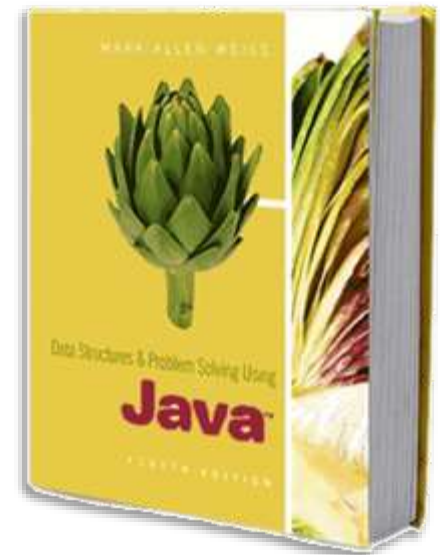
Evtl. kompakt ausdrucken?

2) Lehrbuch



Mark Allen Weiss: *Data Structures & Problem Solving Using Java*, Addison Wesley, 4th Ed., 2010, ISBN-13: 978-0321541406

- **Achtung:** Es gibt diverse „international editions“, bei denen Teile fehlen, z.B. das Kapitel über Simulation (Originalversion hat 985 Seiten)



Materialien und ergänzende Literatur (2)

- Viele **Grundalgorithmen der Informatik** (Sortieren, Suchen etc.) werden hier besonders anschaulich erläutert:
 - Berthold Vöcking, Helmut Alt, Martin Dietzfelbinger, Rüdiger Reischuk, Christian Scheideler, Heribert Vollmer, Dorothea Wagner (Hg.):
Taschenbuch der Algorithmen, Springer 2008



Aus dem Vorwort: „Dieses Buch unternimmt einen umfangreichen Streifzug durch die faszinierende Welt der Algorithmen. Es verlangt keine besonderen Vorkenntnisse, so dass Schülerinnen und Schüler ab der Mittelstufe und auch Informatik-interessierte Laien neue und überraschende Einblicke gewinnen können. In 43 Artikeln von Informatikern, die an Universitäten im In- und Ausland lehren, werden wichtige und besonders elegante Algorithmen anschaulich und verständlich erklärt.“

Dagegen A. Kreye in einem Artikel „Bedeutung von Algorithmen – Neue Weltsprache“ in der Süddeutschen Zeitung vom 22. Juli 2014: „Kurzer Selbstversuch - das «Taschenbuch der Algorithmen», ein Lehrbuch [...]. Die Frustration kommt ähnlich rasch, wie bei Versuchen, Arabisch zu lernen, Chinesisch oder was einen sonst für Flausen reiten, sich aus dem indogermanischen Sprachraum zu entfernen. Da muss man offensichtlich erst einmal die nötigen Hirnwindungen zurechtbiegen, um nur eine Ahnung zu bekommen.“

Materialien und ergänzende Literatur (3)

File Edit View History Bookmarks Tools Help

https://bb.vseth.ethz.ch/frontend.php?action=offe

vseth Bücher suchen

Gross-/Kleinschreibung wird ignoriert und der Suchbegriff wird auch partiell abgeglichen.
Beispiel: *ger* bei Autor gibt Bücher zurück, die von *Egger* oder *Gerd* verfasst wurden.

Titel
Data Structures and Problem Solving Using Java strikte Suche
suchen

Data Structures and Problem Solving Using Java
Autor: Mark A. Weiss
Verlag: Pearson
ISBN: 9781292025766 Auflage: 4

Preis ▲	Zustand ▲	Kommentar	Link
90 CHF	Ausgepackt und doch nie gebraucht	Das Buch befindet sich in neuwertigem Zustand.	mehr Infos
30 CHF	Gebraucht, aber noch in Ordnung	wenige Leuchtstiftmarkierungen, ansonsten sehr guter Zustand	mehr Infos
40 CHF	Ausgepackt und doch nie gebraucht	absolut neuwertig	mehr Infos
50 CHF	Ausgepackt und doch nie gebraucht	Absolut wie neu	mehr Infos

??

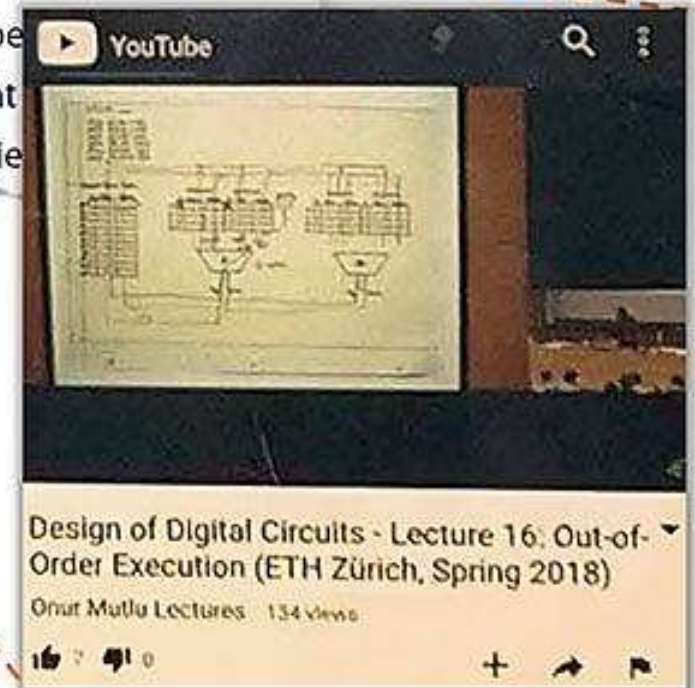
Materialien...?



Vorlesung schauen während dem Radfahren - vermutlich kriminell, dafür zeiteffizient

finden sich sowohl im ETH-Zentrum als auch auf dem Höggerberg die Züri-Velos. Die Bikes gibt es in zwei Varianten: Mit und ohne Hilfsmotor.

ation nützlich sein, oder wenn man noch kurz auf dem Weg zur ETH die verpasste Vorlesung nachsehen möchte. Der Entspe macht zes Lie



Aus: *Visionen 3/2018*, Verein der Informatik-Studierenden an der ETH Zürich, S. 6 - 9

FAQ

- Wieso werden in Informatik I / Informatik II **verschiedene Programmiersprachen** verwendet?
- Was unterscheidet **Teil 2** von **Teil 1** der Vorlesung?
- Wieso gibt es zunächst **keine Beispiellösungen**?
- Wären die Konzerte von Beethoven nicht viel **weniger chaotisch**, wenn erst das Klavier alleine seinen Teil spielen würde und dann der Reihe nach...?

→ Der rote Faden →



Allegro con brio *) Ludwig van Beethoven

Flauto I, II
Oboe I, II
Clarinetto I, II in Si^b / B
Fagotto I, II
Corno I, II in Mi^b / Es
Clarinetto I, II in Do / C
Timpani in Do-Sol / C-G
Violini I
Violini II
Violen
Violoncelli
Bassi

Der rote Faden zur Orientierung



Welt am Sonntag (9.9.12), Art Glazer

Konzepte

Die „Komposition“ der Vorlesung

Java

Korrektheitsnachweis (Invarianten und vollst. Indukt.)
Kryptographie

Java: Basics
Java-Klassen als Datenstrukturen
Klassen-Instanzen und Referenzen

Bäume
Syntaxdiagramme
Rekursiver Abstieg
Infix, Postfix, Operatorbaum, Stack
Codegenerierung, Compiler, Interpreter

Der rote Faden

Verzahnte und **verwobene** Einführung konzeptioneller Aspekte und programmiersprachlicher Konstrukte

Java-VM als Bytecode-Interpreter
Pakete
Klassenhierarchie

Polymorphie


Abstrakte Klassen
Exceptions

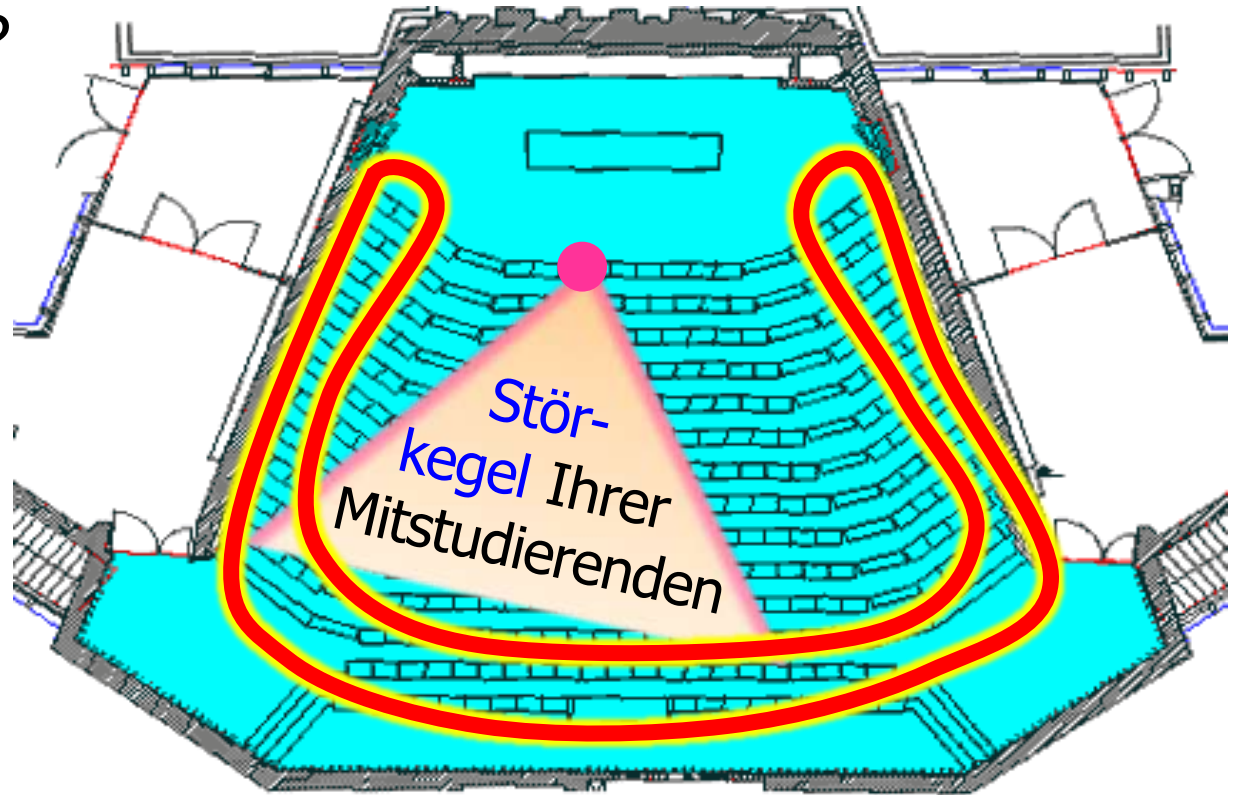
Suchbäume, Sortieren
Backtracking
Spieltheorie, Minimax, Alpha-Beta
Rekursives Problemlösen
Effizienz, O-Notation
Modellierung und Simulation
Heap, Heapsort
Pseudoparallelität

Programmbeispiele dienen gleichzeitig der Einführung programmiersprachlicher Konstrukte und der Illustration von Informatikkonzepten

Threads in Java

Laptops, Tablets, Smartphones, Wearables...?

- Notizen machen?
- ~~■ Web?
 - Google, Wikipedia,...~~
- ~~■ Kommunikation?
 - Whatsapp, Facebook, E-Mail, Twitter,...~~
- Games? 
- ...



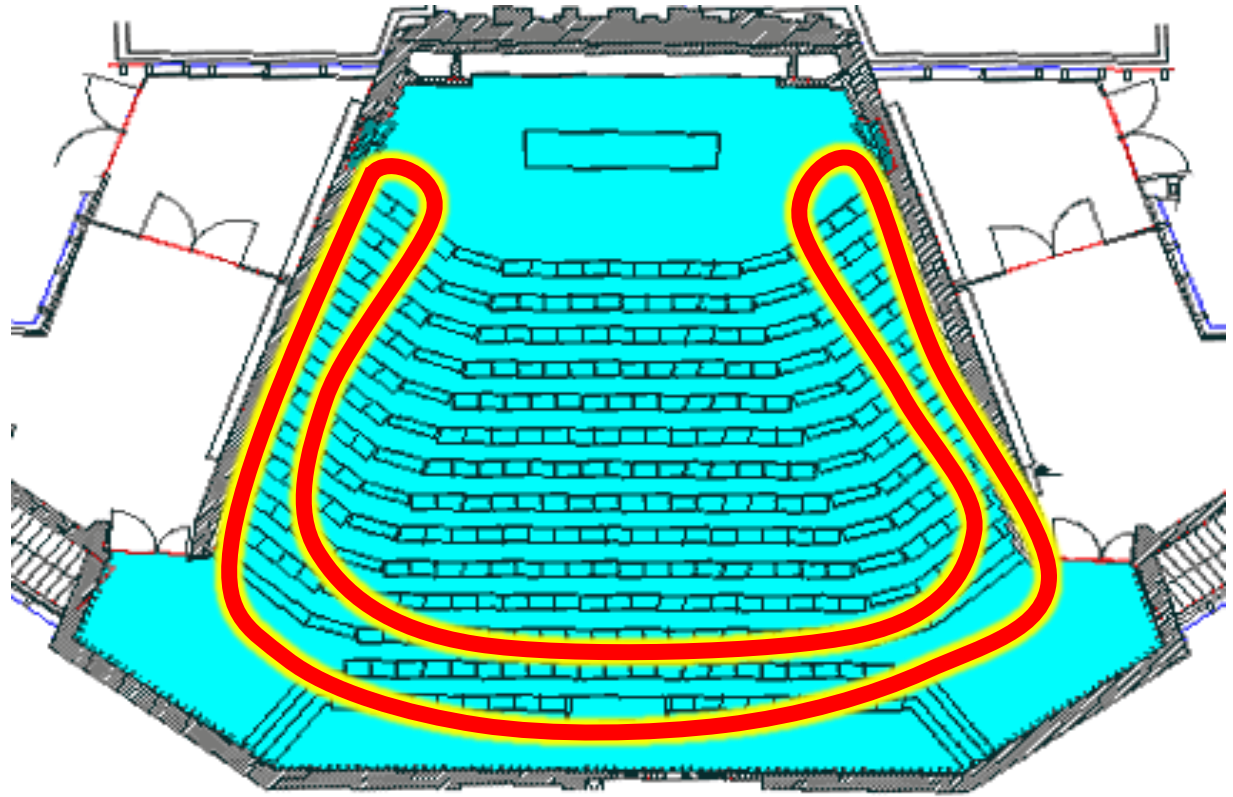
Mit **Laptop** etc. bitte möglichst **Plätze am Rand** nutzen; Tablets sind bzgl. Ablenkung Anderer weniger kritisch

Laptops, Tablets, Smartphones, Wearables...?

■ Notizen machen

"... Recent ... studies ... investigating the differences between note-taking on a laptop and note-taking by hand. While more words were recorded, with more precision, by laptop typists, more ended up being less: regardless of whether a quiz on the material immediately followed the lecture or took place after a week, the pen-and-paper students performed better.

The act of typing effectively turns the note-taker into a transcription zombie." (*The Case for Banning Laptops in the Classroom*, by Dan Rockmore, *The New Yorker*, June 6, 2014, www.newyorker.com/tech/elements/the-case-for-banning-laptops-in-the-classroom)



Mit Laptop etc. bitte möglichst Plätze am Rand nutzen; Tablets sind bzgl. Ablenkung Anderer weniger kritisch

Laptops, Tablets, Smartphones, Wearables...

Henricus de Alemannia hält eine Vorlesung in Bologna

„...haben wir studentische Medien-nutzung anonym und **verdeckt beobachtet**. ...Fokussieren Studierende überwiegend auf **vorlesungsferne Inhalte auf sozialen Webseiten** ...Einzelne Studierende verwenden die gesamte Vorlesungszeit darauf, auf dem Laptop zu **spielen** oder mit Kopfhörer sich Filme und Serien anzusehen. Der Vergleich mit Selbstberichtsdaten zeigt, dass Studierende dabei **unterschätzen, wie häufig sie sich von vorlesungsfernen Inhalten ablenken lassen** und wie ungünstig sich diese Ablenkungen **auf ihren Lernerfolg auswirken**.

Verschiedene Motive existieren für den ablenkenden und daher nachteiligen Gebrauch mobiler Geräte in Vorlesungen. Eines davon ist die **Angst, etwas zu versäumen**, was sich freilich nicht auf die Vorlesungsinhalte sondern auf soziale Online-Informationen bezieht. Die stets verfügbaren Online-Ablenkungen scheinen kleine Belohnungen darzustellen, und viele Studierende haben es sich **zur Gewohnheit gemacht, hochfrequent Online-Informationen abzurufen**.



Armin Weinberger: Vorlesung und Ablenkung. Forschung und Lehre 10, Oktober 2014, S. 838-839

Wissenschaftliches Ethos auch bei den Übungen

- **Alle Quellen / Ko-Autoren** zitieren
 - Web, Bücher
 - Studierende anderer Übungsgruppen (z.B. bei Lerngruppen)
 - Studierende früherer Jahrgängen
 - ...
- **Nur Eigenbeitrag** wird bewertet
- **Unehrliches Handeln** führt zumindest zur Annullierung des gesamten Übungsblatts
- Vgl. auch **Zitierhinweise** bei
 - www.ethz.ch/studierende/de/studium/leistungskontrollen/plagiate.html

