

Algorithmen und Datenstrukturen

A1. Organisatorisches

Marcel Lüthi and Gabriele Röger

Universität Basel

23. Februar 2022

Algorithmen und Datenstrukturen

23. Februar 2022 — A1. Organisatorisches

A1.1 Organisatorisches

A1.2 Über diese Vorlesung

A1.1 Organisatorisches

Personen: Dozenten



Marcel Lüthi



Gabriele Röger

Dozenten

Dr. Marcel Lüthi

- ▶ **E-Mail:** `marcel.luethi@unibas.ch`
- ▶ **Büro:** Raum 04.002, Spiegelgasse 1

Dr. Gabriele Röger

- ▶ **E-Mail:** `gabriele.roeger@unibas.ch`
- ▶ **Büro:** Raum 04.005, Spiegelgasse 1

Personen: Tutoren



Max Jappert

- ▶ **E-Mail:** `max.jappert@unibas.ch`
- ▶ Di 14:15-16:00

Reto Krummenacher

- ▶ **E-Mail:** `reto.krummenacher@unibas.ch`
- ▶ Fr 14:15-16:00

Colin Fingerlin

- ▶ **E-Mail:** `colin.fingerlin@unibas.ch`
- ▶ Mi, 10:15-12:00

Zeit & Ort

Vorlesungen

- ▶ Mi 14:15-16:00 Uhr, Biozentrum, Hörsaal U1.131
- ▶ Do 14:15-16:00 Uhr, Biozentrum, Hörsaal U1.141

Übungen

- ▶ Di 14:15-16:00, Pharmazentrum U1075
- ▶ Mi 10:15-12:00, Spiegelgasse 1, U1.001
- ▶ Fr 14:15-16:00, **Online oder Pharmazentrum** U1075

Erster Übungstermin 25. Februar / 1. März / 2. März

Vorlesung im Web

Vorlesungsseite

`https://dmi.unibas.ch/de/studium/
computer-science-informatik/lehrangebot-fs22/
vorlesung-algorithmen-und-datenstrukturen/`

- ▶ Vorlesungsbeschreibung
- ▶ Folien
- ▶ Zusatzmaterial (nicht prüfungsrelevant)

Anmeldung:

- ▶ `https://services.unibas.ch/`
- ▶ Bitte registrieren Sie sich gleich heute, um alle kursrelevanten Informationen zu erhalten.

Vorlesungsmaterialien

Vorlesungsmaterialien:

- ▶ Vorlesungsfolien (online)
- ▶ Lehrbuch
- ▶ vertiefendes Material **auf Anfrage**

Lehrbuch



Algorithmen
von Robert Sedgwick und Kevin Wayne
(Pearson Verlag, 4. Auflage)

Weitere nützliche Ressourcen

- ▶ Seite zum Buch: Algorithms, 4th edition:
<https://algs4.cs.princeton.edu/home/>
- ▶ Youtube (Suchbegriff: Robert Sedgewick algorithms)
- ▶ Data Structures and Algorithms – The Basic Toolbox
von Kurt Mehlhorn und Peter Sanders (Springer Verlag)
<https://people.mpi-inf.mpg.de/~mehlhorn/Toolbox.html>
- ▶ Google, Wikipedia, ...

Zielgruppe

Zielgruppe:

- ▶ Bachelor Informatik (ab 2. Semester)
- ▶ Bachelor Computational Sciences (ab 2. Semester)
- ▶ Alle Studierenden mit Programmierkenntnissen sind herzlich willkommen.

Voraussetzungen:

- ▶ Programmierkenntnisse (Idealerweise Java)

Programmiersprachen

- ▶ Vorlesung: Hauptsächlich Python
→ Vorteil: Kompakt und direkt, ideal für kleine Programme
- ▶ Übungen: Java oder Python (nach Ankündigung)



Es werden keine Python-Kenntnisse vorausgesetzt!

Übungen

Übungsaufgaben:

- ▶ Hausaufgaben (Theorie + Praxis)

Übungstermine:

- ▶ Vorbesprechung der Hausaufgaben
- ▶ Beantwortung von Fragen zum aktuellen Blatt
- ▶ Technische Hilfestellung (Java/Python, Programmierumgebung)
- ▶ Nachbesprechung der Übungen
- ▶ Teilnahme freiwillig - **aber sehr empfohlen.**

Übungen: Hausaufgaben

Hausaufgaben:

- ▶ Aufgaben ab Donnerstagabend auf Adam verfügbar.
- ▶ Bearbeitung in Gruppen empfohlen
 - ▶ Gruppenmitglieder sollen in derselben Übungsgruppe sein
- ▶ Abgabe freitags in Folgewoche (23:59) auf Adam
- ▶ Besprechung und **individuelles Feedback** in Übungsgruppe

Wahl der Übungsgruppe

- ▶ Bitte geben Sie Ihre Präferenzen an.
- ▶ Beachten Sie
 - ▶ Gruppenmitglieder müssen in derselben Gruppe sein.
 - ▶ Freitag ist entweder ganz Online oder ganz in Präsenz, je nach vorheriger Abstimmung.

Link zu google form

<https://bit.ly/adfs2022>

Übungen: Zwischenprüfungen

Die Übungen sind eigene Veranstaltung und geben 2 Kreditpunkte!

Erhalt der Kreditpunkte

- ▶ Zwischenprüfungen am 7. April und 19. Mai
 - ▶ Ort: Üblicher Vorlesungsraum (Biozentrum)
- ▶ Bewertung: Kumulierte Punktzahl aus beiden Prüfungen entscheidet über Pass/Fail

Discord-Server

- ▶ Fragen können auf dem Vorlesungs-Discord-Server diskutiert werden.
 - ▶ Idee: Kursteilnehmer helfen sich gegenseitig.
 - ▶ Dozierende können auf Anfrage mithelfen.

Erklären und helfen heisst lernen!

Examen

- ▶ **Examen:** schriftliche Prüfung der **Hauptvorlesung**
- ▶ 6 Kreditpunkte, bewertet mit Noten 1.0–6.0
- ▶ voraussichtlich Mo, 13. Juni 2022, 14-16 Uhr
- ▶ Separate **Anmeldung zum Examen:** 4.–19. April
↪ siehe auch <https://philnat.unibas.ch/de/examen/>
- ▶ Keine Zulassungsbeschränkung
- ▶ Note basiert nur auf Klausur
- ▶ Bei Nichtbestehen **ein** Wiederholungsversuch im FS 2023!

Vorlesung am 2. Juni: **Q&A zu gesamter Vorlesung und Prüfung**

Laptops

Miniübungen während der Vorlesung: Bitte Laptops mitbringen.

Aber Vorsicht:

Research Article

Logged In and Zoned Out: How Laptop Internet Use Relates to Classroom Learning

**Susan M. Ravizza, Mitchell G. Uitvlugt, and
Kimberly M. Fenn**

Department of Psychology, Michigan State University, East Lansing

Abstract

Laptop computers are widely prevalent in university classrooms. Although laptops are a valuable tool, they offer access to a distracting temptation: the Internet. In the study reported here, we assessed the relationship between classroom performance and actual Internet usage for academic and nonacademic purposes. Students who were enrolled in an introductory psychology course logged into a proxy server that monitored their online activity during class. Past



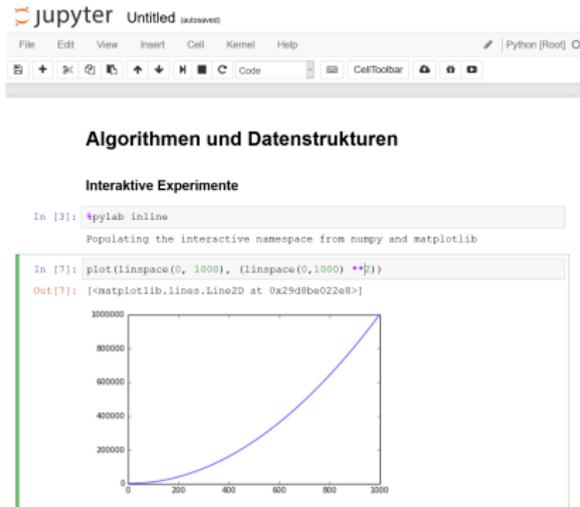
Psychological Science
2017, Vol. 28(2), 171–180
© The Author(s) 2017
Reprints and permissions:
sagepub.com/journalsPermissions.nav
DOI: 10.1177/0956797616667314
www.psychologicalscience.org/PS
The SAGE logo consists of a stylized 'S' inside a circle, followed by the word 'SAGE' in a bold, uppercase, sans-serif font.

Jupyter-Notebooks

Web-basierte interaktive Arbeitsumgebung für Python

Nutzung von Jupyter-Notebooks:

- ▶ Erklärungen zu Algorithmen
- ▶ Implementation von Algorithmen
 - ▶ Experimentieren und Lernen zuhause
- ▶ Miniübungen während der Vorlesung



Fragen zur Organisation

Fragen?

A1.2 Über diese Vorlesung

Algorithmen und Datenstrukturen

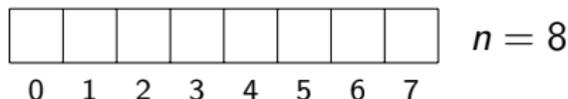
- ▶ Bestimmte Grundbausteine benötigt man immer wieder bei Programmierprojekten, z.B.
 - ▶ Sortierverfahren
 - ▶ Suchbäume
 - ▶ Prioritätswarteschlangen
 - ▶ kürzeste Pfade in Graphen
 - ▶ ...
- ▶ Wird oftmals durch Bibliotheken fertig bereitgestellt.
- ▶ Hier lernen Sie ...
 - ▶ wie das alles intern funktioniert.
 - ▶ wie man den richtigen Baustein auswählt.
 - ▶ Tricks und Kniffe für effiziente Programme.
- ▶ Methoden unabhängig von konkreter Programmiersprache

Beispiel: Sortieralgorithmen

- ▶ Aufgabe: Bringe Sequenz von Elementen in aufsteigende Reihenfolge, z.B.
Eingabe [5, 9, 3, 5] → Ausgabe [3, 5, 5, 9]
- ▶ 1960er Jahre (und noch lange danach): ein Viertel der kommerziell verbrauchten Rechenzeit für Sortiervorgänge
- ▶ Naiver Algorithmus: **Selectionsort**



Selectionsort: Informell



- ▶ Finde kleinstes Element an Positionen $0, \dots, n - 1$ und tausche es an Position 0
- ▶ Finde kleinstes Element an Positionen $1, \dots, n - 1$ und tausche es an Position 1
- ▶ ...
- ▶ Finde kleinstes Element an Positionen $n - 2, \dots, n - 1$ und tausche es an Position $n - 2$

Selectionsort: Beispiel

3	7	2	9	7	1	4	5
---	---	---	---	---	---	---	---

1	7	2	9	7	3	4	5
---	---	---	---	---	---	---	---

1	2	7	9	7	3	4	5
---	---	---	---	---	---	---	---

1	2	3	9	7	7	4	5
---	---	---	---	---	---	---	---

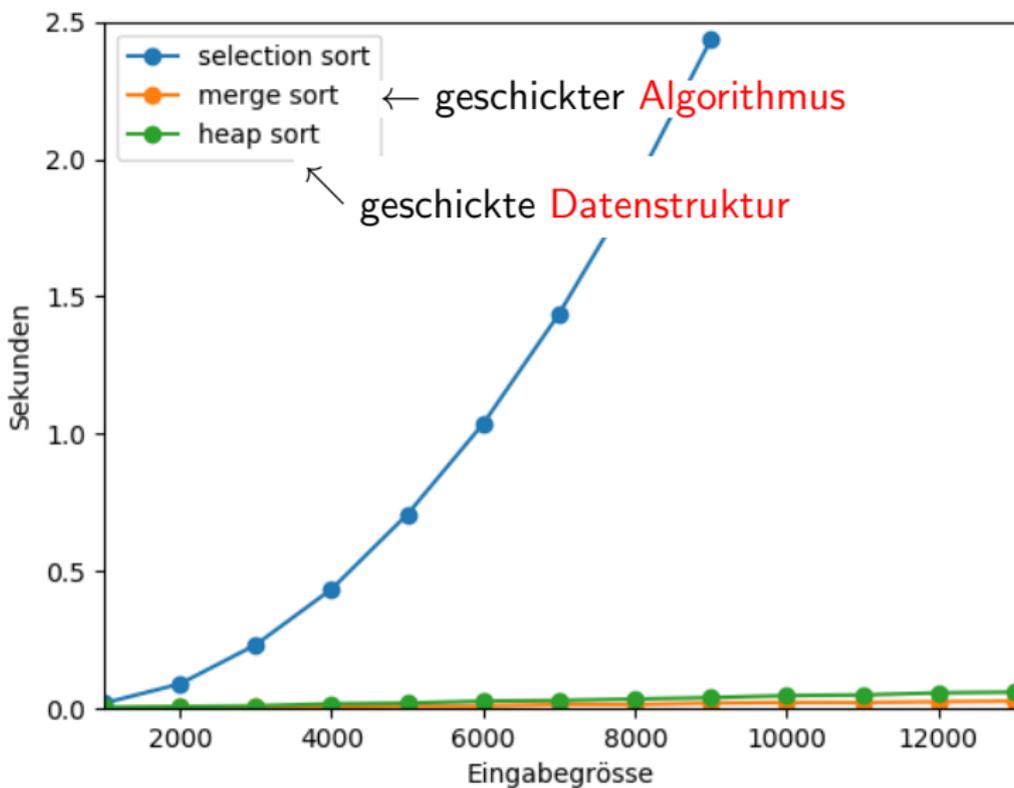
1	2	3	4	7	7	9	5
---	---	---	---	---	---	---	---

1	2	3	4	5	7	9	7
---	---	---	---	---	---	---	---

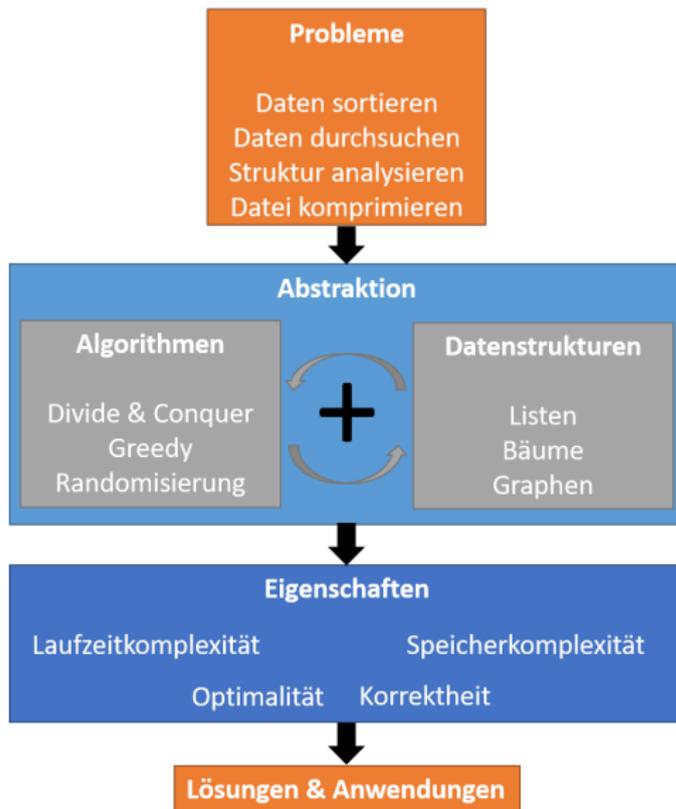
1	2	3	4	5	7	9	7
---	---	---	---	---	---	---	---

1	2	3	4	5	7	7	9
---	---	---	---	---	---	---	---

Sortieralgorithmen: Laufzeit



Der Kurs Algorithmen und Datenstrukturen



Inhalt dieser Veranstaltung

