

# Algorithmen und Datenstrukturen

## A1. Organisatorisches

Marcel Lüthi and Gabriele Röger

Universität Basel

22. Februar 2023

# Algorithmen und Datenstrukturen

22. Februar 2023 — 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](mailto:marcel.luethi@unibas.ch)
- ▶ **Büro:** Raum 04.009, Spiegelgasse 1

Dr. Gabriele Röger

- ▶ **E-Mail:** [gabriele.roeger@unibas.ch](mailto:gabriele.roeger@unibas.ch)
- ▶ **Büro:** Raum 04.005, Spiegelgasse 1

# Personen: Tutoren



## Kian Hunziker

- ▶ **E-Mail:** [kian.hunziker@unibas.ch](mailto:kian.hunziker@unibas.ch)
- ▶ Mi, 10:15-12:00

## Linus Petrucci

- ▶ **E-Mail:** [l.petrucci@unibas.ch](mailto:l.petrucci@unibas.ch)
- ▶ Fr 14:15-16:00

## Sylvain Rousselle

- ▶ **E-Mail:** [sylvain.rousselle@unibas.ch](mailto:sylvain.rousselle@unibas.ch)
- ▶ Di 14:15-16:00

# 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, Spiegelgasse 1, U1.001

Erster Übungstermin 24. Februar / 7. März / 8. März

# Zielgruppe

## Zielgruppe:

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

## Voraussetzungen:

- ▶ Programmierkenntnisse (idealerweise Java; Python geht auch)

# Materialien

- ▶ **ADAM:** <https://adam.unibas.ch/>  
Zentraler Startpunkt, Übungen, Lernmodule  
(Datenstrukturen)
- ▶ **Webseite:** Vorlesungsbeschreibung & Material (Folien,  
Jupyter-Notebooks, ...)
- ▶ **Discord:** für Interaktion untereinander
  - ▶ Idee: Kursteilnehmer helfen sich gegenseitig.
  - ▶ Dozierende können auf Anfrage mithelfen.

# Lehrbuch

## Lehrbuch



Algorithmen  
von Robert Sedgewick 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, ...

# 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:

- ▶ Einführung neues Übungsblatt
- ▶ Beantwortung von Fragen zum aktuellen Blatt und zur Vorlesung
- ▶ Nachbesprechung letztes Blatt
- ▶ Technische Hilfestellung (Java/Python, Programmierumgebung)
- ▶ Inhaltliche Hilfestellung zum aktuellen Blatt
- ▶ 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
  - ▶ Tragen Sie sich gemeinsam mit Ihren Gruppenmitgliedern ein.
  - ▶ Alle Gruppenmitglieder kommen in dieselbe Übungsstunde.

## Formular



<https://tny.lv/ads23>

# Neues Format seit 2022

- ▶ **Zuvor:** 8 ECTS für Vorlesung und Übung
- ▶ **Neu:** 6 ECTS Hauptvorlesung + 2 ECTS Übung
- ▶ Separate Anmeldung und Bewertung
- ▶ Können und sollen gleichzeitig belegt werden.

## Anmeldung

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

# Bewertung der Übung

## Erhalt der 2 Kreditpunkte

- ▶ Zwischenprüfungen am 5. April und 11. Mai
  - ▶ Ort: Üblicher Vorlesungsraum (Biozentrum)
- ▶ Bewertung: Kumulierte Punktzahl aus beiden Prüfungen entscheidet über Pass/Fail

# Bewertung der Hauptvorlesung

- ▶ **Examen:** schriftliche Prüfung
- ▶ 6 Kreditpunkte, bewertet mit Noten 1.0–6.0
- ▶ voraussichtlich Mo, 12. Juni 2023, 14-16 Uhr
- ▶ Separate **Anmeldung zum Examen:** 3.–17. April  
~~ siehe auch <https://philnat.unibas.ch/de/examen/>
- ▶ Keine Zulassungsbeschränkung
- ▶ Note basiert nur auf Klausur
- ▶ Bei Nichtbestehen **ein** Wiederholungsversuch im FS 2024!

Vorlesung am 1. 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



Psychological Science  
2017, Vol. 28(2) 171–190  
© The Authors 2017  
Reprints and permission:  
[sagepub.com/journalsPermissions.nav](http://sagepub.com/journalsPermissions.nav)  
DOI: 10.1177/0956797616677314  
[www.psychologicalscience.org/PS](http://www.psychologicalscience.org/PS)  
SAGE

### 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

# 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

The screenshot shows a Jupyter Notebook window titled "jupyter Untitled (unsaved)". The menu bar includes File, Edit, View, Insert, Cell, Kernel, and Help. The toolbar includes icons for file operations, cell selection, and code execution. A tab labeled "Python [Root]" is selected. The main area contains two code cells and one output cell. The first cell is titled "Algorithmen und Datenstrukturen" and "Interaktive Experimente". It contains the command "%pylab inline" and outputs "Populating the interactive namespace from numpy and matplotlib". The second cell contains the command "plot(linspace(0, 1000), (linspace(0,1000) \*\*2))" and its output, which is a blue line plot showing a quadratic relationship between the x and y axes, ranging from 0 to 1,000,000.

```
In [3]: %pylab inline
Populating the interactive namespace from numpy and matplotlib
In [7]: plot(linspace(0, 1000), (linspace(0,1000) **2))
Out[7]: [

Algorithmen und Datenstrukturen



Interaktive Experimente



```
In [3]: %pylab inline
Populating the interactive namespace from numpy and matplotlib
In [7]: plot(linspace(0, 1000), (linspace(0,1000) **2))
Out[7]: [
```


```

# Fragen zur Organisation

Fragen?

## A1.2 Über diese Vorlesung

# Algorithmen und Datenstrukturen

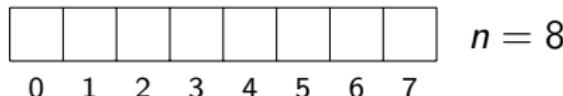
- ▶ Bestimmte Grundbausteine benötigt man immer wieder bei Programmierprojekten, z.B.
  - ▶ Sortierverfahren
  - ▶ Suchbäume
  - ▶ Prioritätswarteschlangen
  - ▶ kürzeste Pfade in Graphen
  - ▶ ...
- ▶ Wird oftmals durch Bibliotheken fertig bereitgestellt.
- ▶ Hier lernen Sie ...
  - ▶ wie das alles 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 |
|---|---|---|---|---|---|---|---|

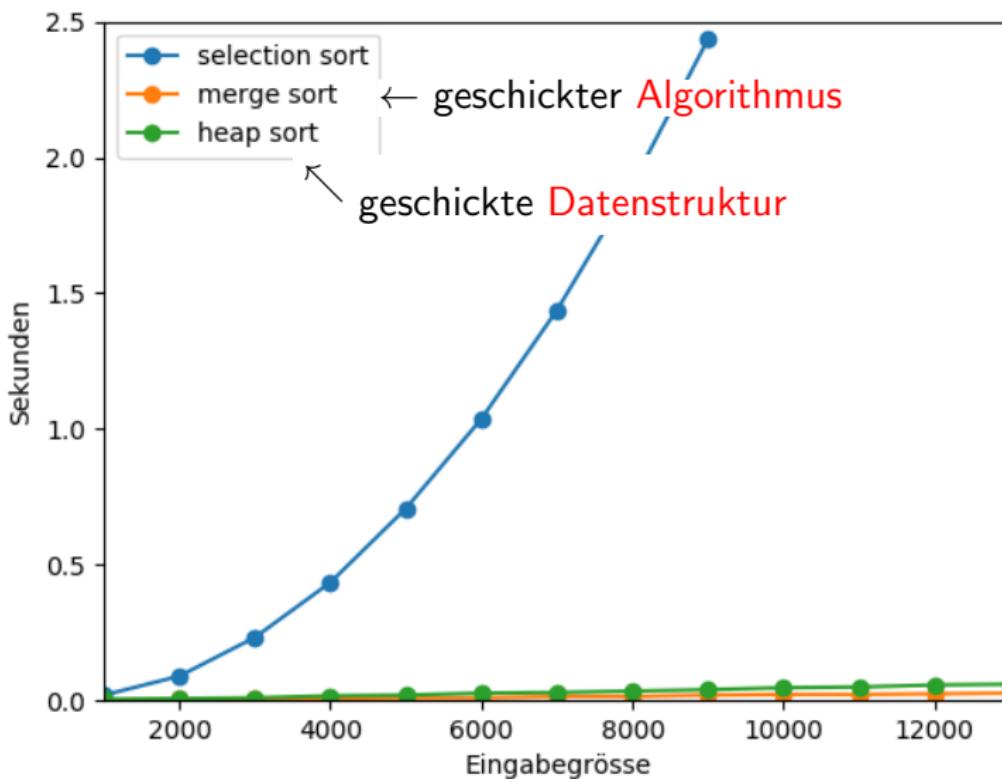
|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 7 | 7 | 9 | 5 |
|---|---|---|---|---|---|---|---|

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

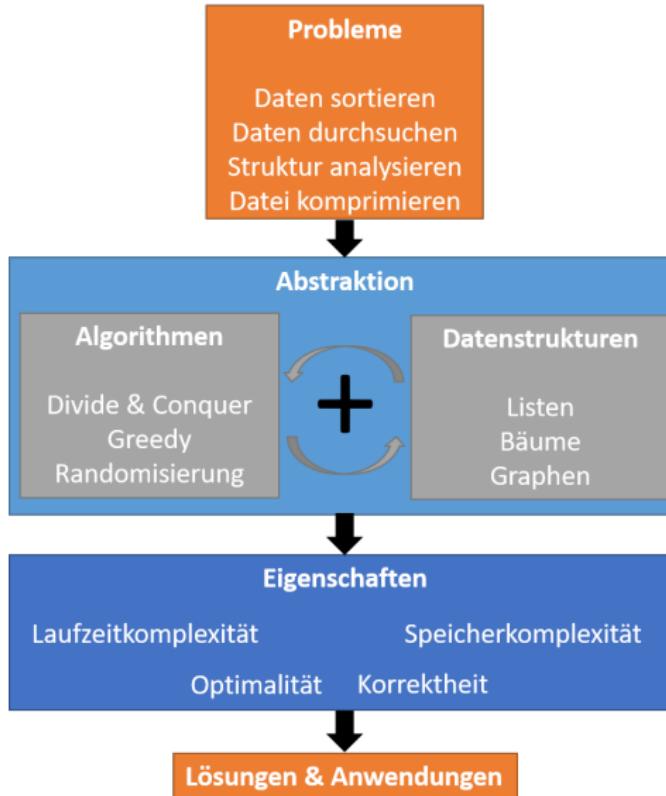
|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 7 | 9 | 7 |
|---|---|---|---|---|---|---|---|

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

# Sortieralgorithmen: Laufzeit



# Der Kurs Algorithmen und Datenstrukturen



# Inhalt dieser Veranstaltung

