Algorithms and Data Structures

A1. Organizational Matters

Gabriele Röger and Patrick Schnider

University of Basel

February 18, 2026

Organizational Matters

People

Gabriele Röger

Patrick Schnider

Lecturers

Gabi Röger

- email: gabriele.roeger@unibas.ch
- office: room 04.005, Spiegelgasse 1

Patrick Schnider

- email: patrick.schnider@unibas.ch
- office: room 04.009, Spiegelgasse 1

People

Tutors

Pascal von Fellenberg (pascal.vonfellenberg@unibas.ch)

- Tuesday, 14.15-16.00, Pharmazentrum, U1075

Julia Sauer (julia.sauer@stud.unibas.ch)

- Wednesday, 10.15-12.00, Pharmazentrum, U1075

Ramakrishnan Mani (r.mani@stud.unibas.ch)

- Friday, 14.15-16.00, Pharmazentrum, U1075

Time & Place

Lectures

- Wednesday: 14:15–16:00, Biozentrum, lecture hall U1.131
- Thursday: 14:15–16:00, Alte Universität, lecture hall -101

Exercise Sessions (starting February 20, March 3/4)

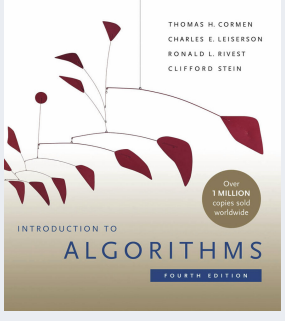
- Tuesday, 14.15-16.00, Pharmazentrum, U1075
- Wednesday, 10.15-12.00, Pharmazentrum, U1075
- Friday, 14.15-16.00, Pharmazentrum, U1075

Resources

- Adam: central starting point and exercises
  `https://adam.unibas.ch/`
- Website: course information, slides, notebooks
- Discord: for your interaction with each other
  - Idea: course participants help each other.
  - Lecturers and tutors can help by request.
  - Feel free to use a pseudonym.

Textbook

Textbook



Introduction to Algorithms
by Thomas H. Cormen, Charles E. Leiserson,
Ronald L. Rivest and Clifford Stein
(MIT Press, Fourth Edition)

Programming Languages

- Lectures: Mostly Python
  $\rightarrow$ Advantage: compact and direct, ideal for smaller programs
- Exercises: Java or Python (indicated on exercise sheet)

We don't require any previous knowledge about Python!

Exercises

Exercise sheets (homework assignments):

- theoretical and programming exercises
- on ADAM every Thursday evening
- may be solved in groups (we recommend groups of 2-3)
- group members should be in same exercise group
- due Friday the following week (23:55)
  (upload to Adam at `https://adam.unibas.ch/`)
- discussion and individual feedback in exercise meeting

Exercises

Exercise sessions:

- introduction of/questions about the current exercise sheet
- discussion of previous exercise sheet (common problems)
- questions about the course
- if time: work on the homework assignment
  - support with the current exercise sheet
  - technical support (Java/Python, programming environment)
- participation voluntary but highly recommended

Exercises

Exercise sessions:

- introduction of/questions about the current exercise sheet
- discussion of previous exercise sheet (common problems)
- questions about the course
- if time: work on the homework assignment
  - support with the current exercise sheet
  - technical support (Java/Python, programming environment)
- participation voluntary but highly recommended

important: please fill in the survey on ADAM for the group assignment until tomorrow 14:00 (February 19).

- One registration per team (please list all names).
- All team members will be in the same exercise session.

Course Format

- 6 ECTS main course + 2 ECTS exercises
- separate enrolment and evaluation
- can and should be taken in parallel

Enrolment

- `https://services.unibas.ch/`
- register today for the course, so that you get all relevant emails and access to the ADAM workspace
- enrolment for exercise after we made the group assignment

Prerequisites

- basic programming skills (ideally Java or Python)

Evaluation of Main Course (6 CP)

- written exam, 6 ECTS credits, graded 1-6
- 10 June 2026, 14:00-16:00,
  Biozentrum, rooms U1.101 and U1.131
- admission to exam: no prerequisites
- must register for exam during March 30 – April 13
  $\rightsquigarrow$ see `https://philnat.unibas.ch/de/examen/`
- grade for course determined exclusively by the exam
- if you fail: one repeat attempt (within one year)

Evaluation of Main Course (6 CP)

- written exam, 6 ECTS credits, graded 1-6
- 10 June 2026, 14:00-16:00,
  Biozentrum, rooms U1.101 and U1.131
- admission to exam: no prerequisites
- must register for exam during March 30 – April 13
  $\rightsquigarrow$ see `https://philnat.unibas.ch/de/examen/`
- grade for course determined exclusively by the exam
- if you fail: one repeat attempt (within one year)

Last lecture (May 28): Q&A session for exam preparation

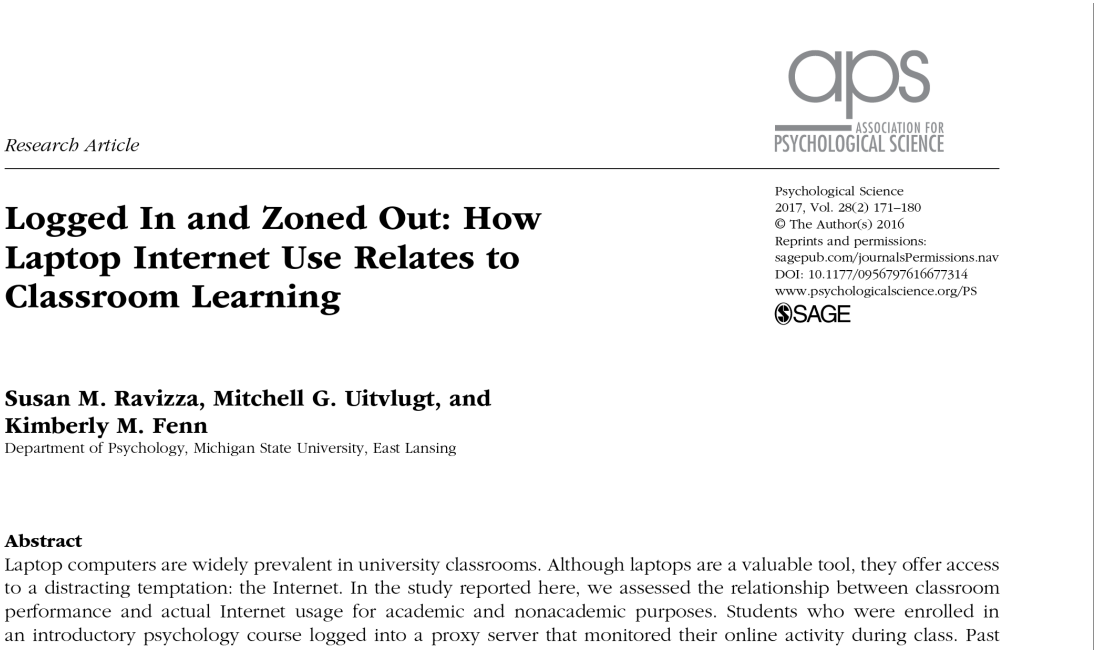
Evaluation of Exercises (2 CP)

- midterm exams on April 8 and May 6
- in the usual lecture hall (Biozentrum)
- pass/fail evaluation based on the accumulated marks from the midterm exams

Laptops

Small exercises during the lecture: please bring your laptop.

But stay focused:



Research Article

aps ASSOCIATION FOR PSYCHOLOGICAL SCIENCE

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

Psychological Science
2017, Vol. 28(2) 171–180

DOI: 10.1177/0956797616677314
www.psychologicalscience.org/PS
SAGE

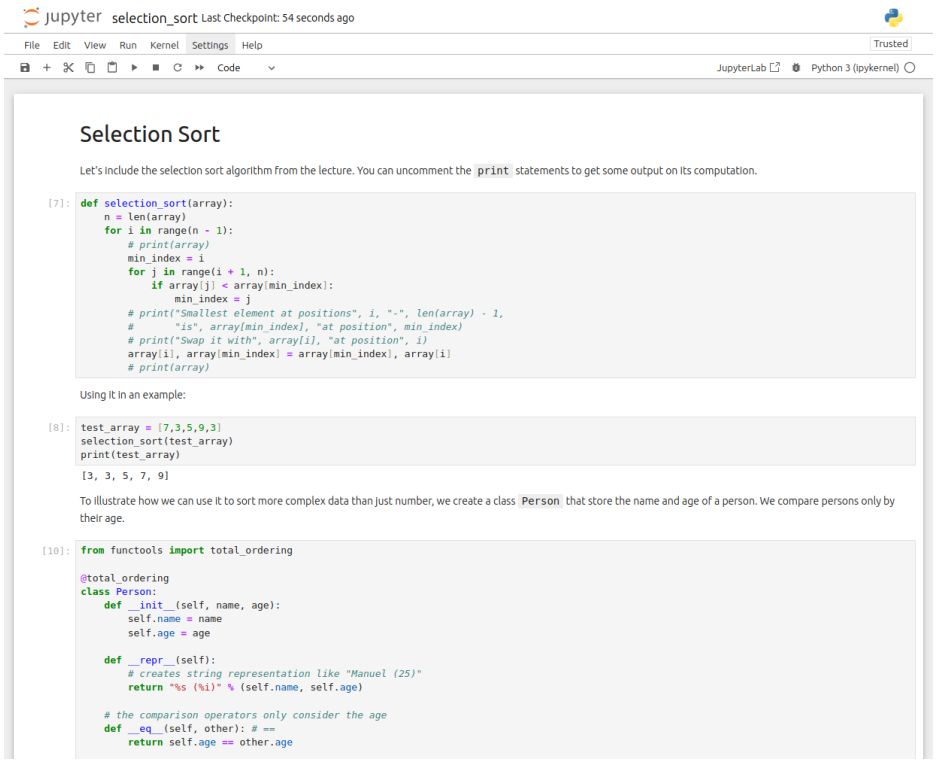
Susan M. Ravizza, Mitchell G. Uitvlugt, and Kimberly M. Fenn
Department of Psychology, Michigan State University, East Lansing

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

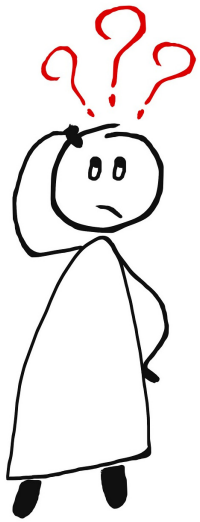
Jupyter Notebooks

web-based interactive computational environment for Python (and some other languages)

- illustrating algorithms and concepts
- implementing algorithms for experimenting and studying at home
- small exercises during the lecture



Questions on Organization



Questions?

About this Course

Algorithms and Data Structures

- some basic building blocks are needed again and again in programming projects, e.g.
  - sorting algorithms
  - search trees
  - priority queues
  - shortest path in a graph
  - ...
- often provided by libraries

Algorithms and Data Structures

- some basic building blocks are needed again and again in programming projects, e.g.
  - sorting algorithms
  - search trees
  - priority queues
  - shortest path in a graph
  - ...
- often provided by libraries
- here you learn ...
  - how all this works internally.
  - how to select suitable building blocks.
  - tricks to achieve efficient programs.

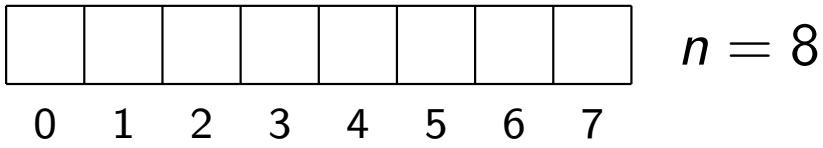
Algorithms and Data Structures

- some basic building blocks are needed again and again in programming projects, e.g.
  - sorting algorithms
  - search trees
  - priority queues
  - shortest path in a graph
  - ...
- often provided by libraries
- here you learn ...
  - how all this works internally.
  - how to select suitable building blocks.
  - tricks to achieve efficient programs.
- independent of specific programming language

Example: Algorithms for Sorting

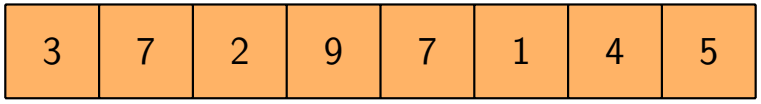
- task: sort a sequence of elements in increasing order, e.g.
  input [5, 9, 3, 5] $\rightarrow$ result [3, 5, 5, 9]
- 1960s (and a long time afterwards): a quarter of all commercial computation time used for sorting
- naive algorithm: selection sort

Selection Sort: Informally

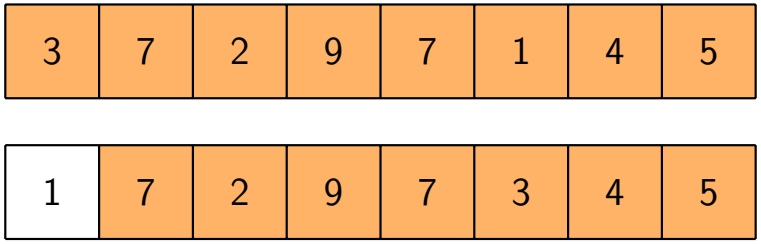


- identify smallest element at positions $0, \ldots, n-1$ and swap it to position 0
- identify smallest element at positions $1, \ldots, n-1$ and swap it to position 1
- ...
- identify smallest element at positions $n-2, n-1$ and swap it to position n - 2

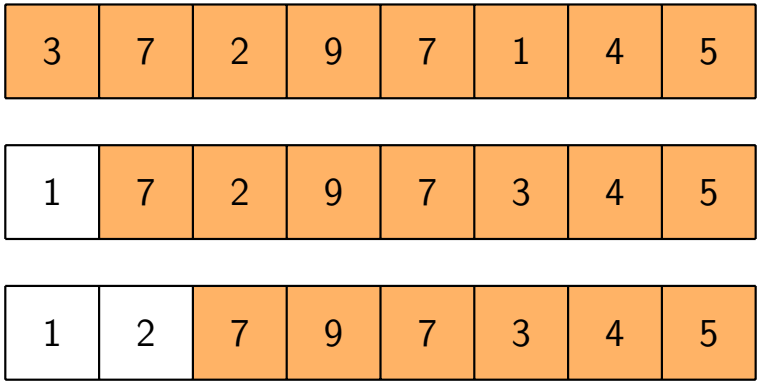
Selection Sort: Example



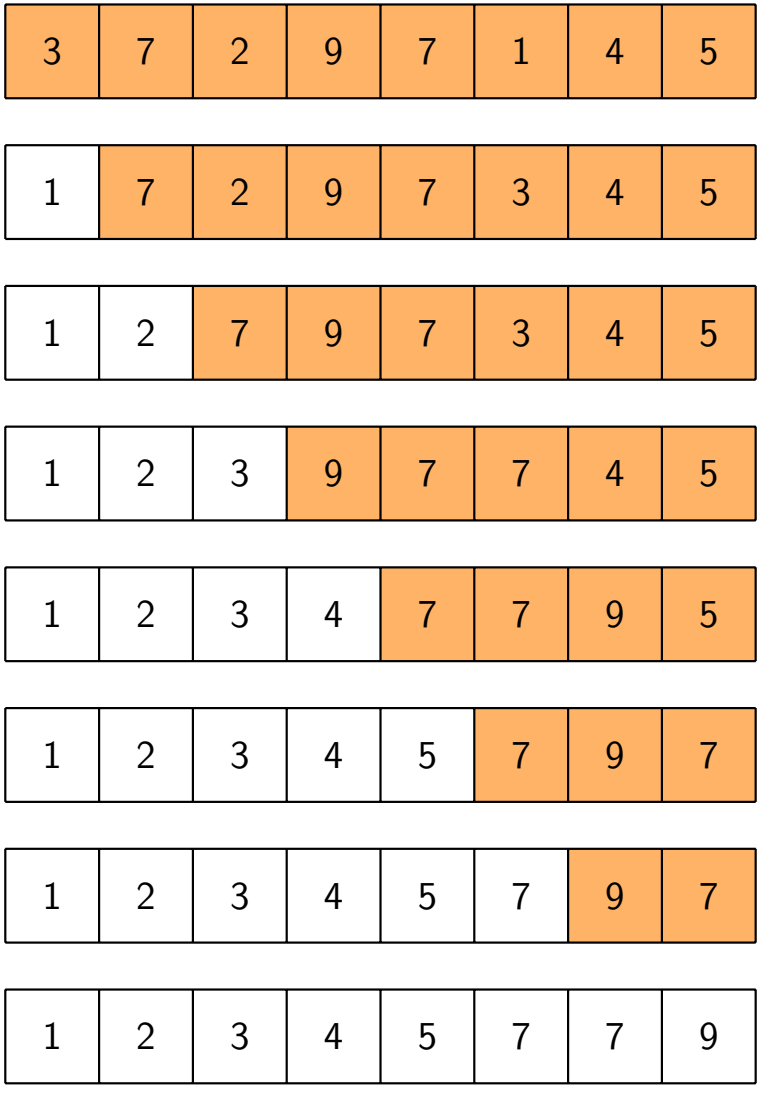
Selection Sort: Example



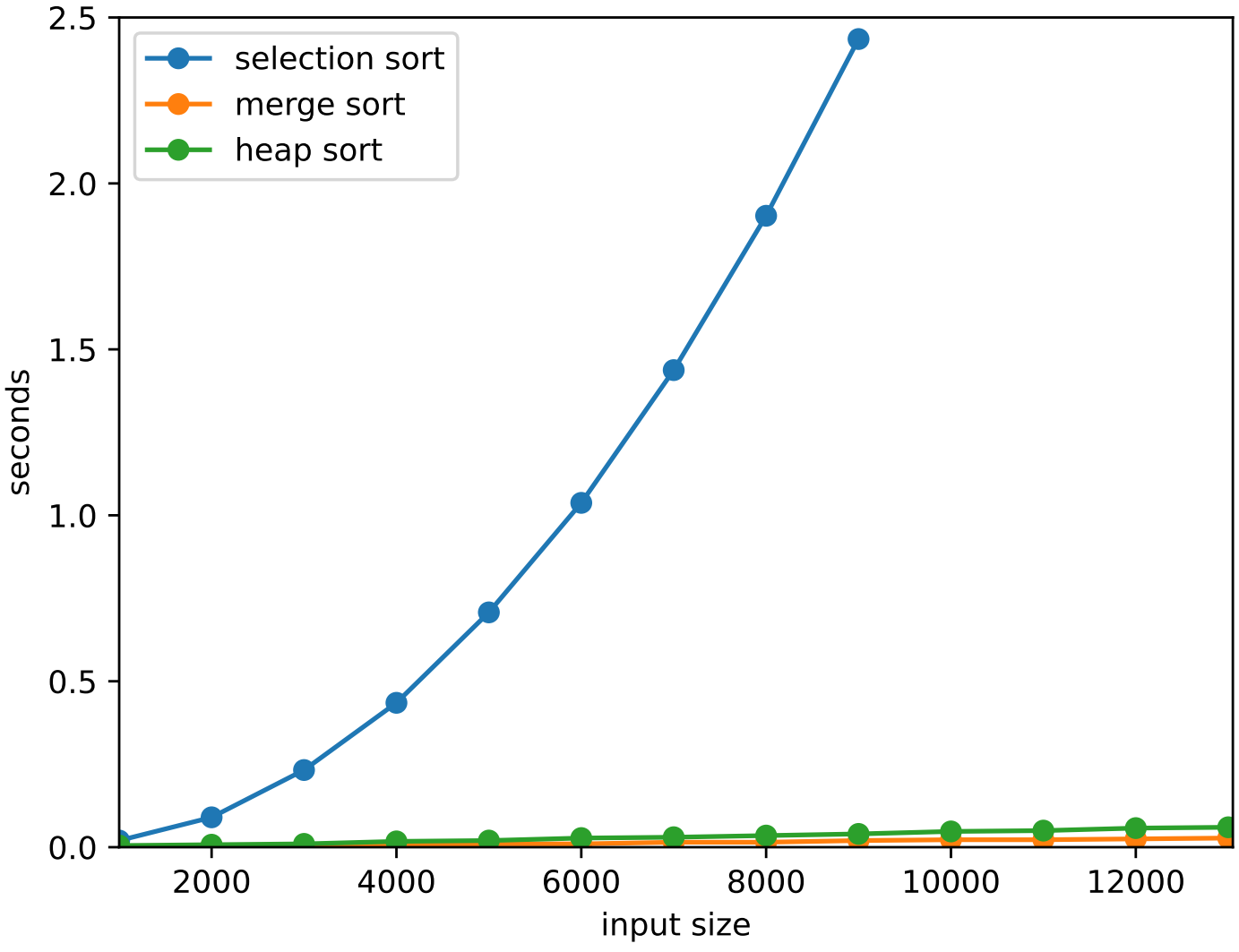
Selection Sort: Example



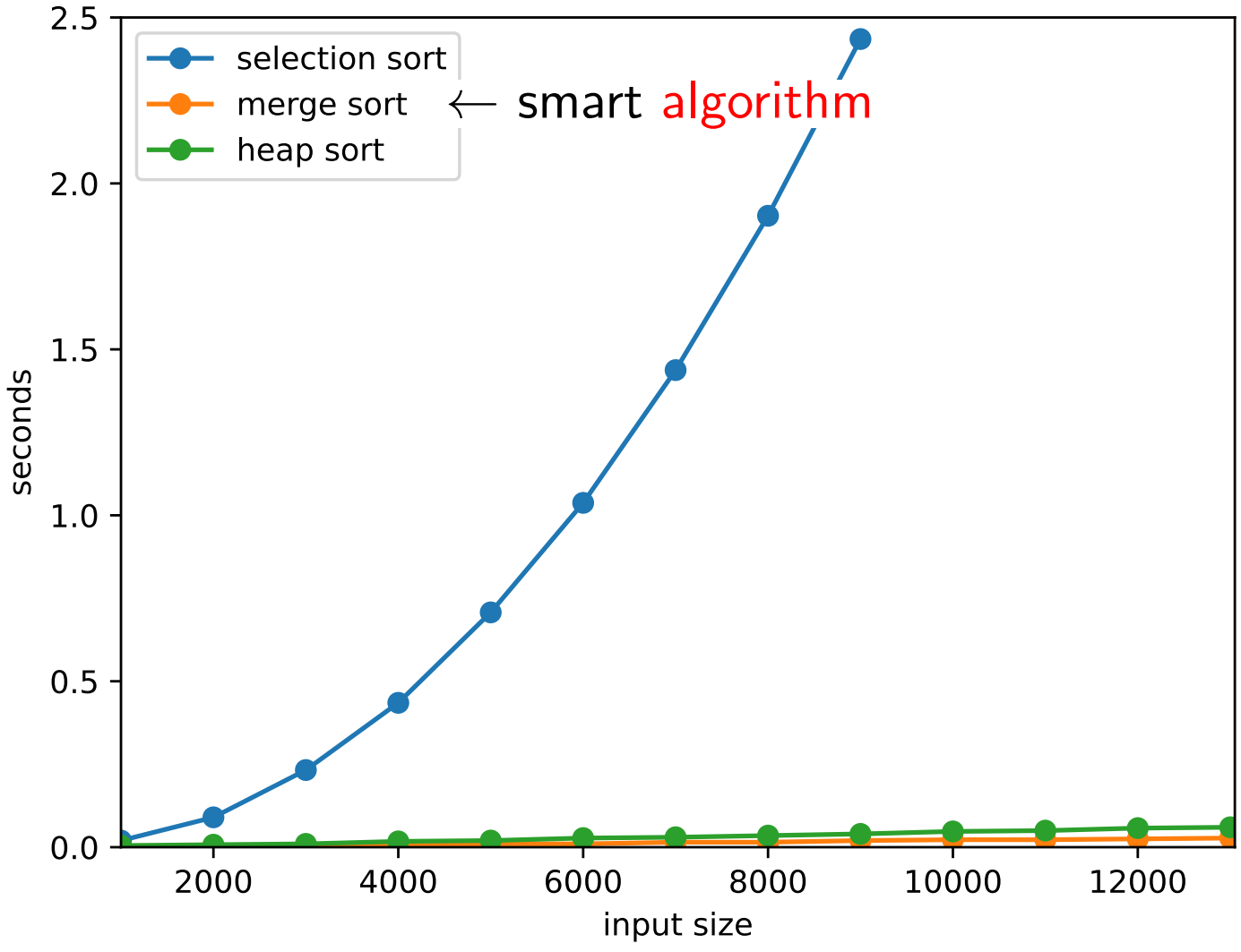
Selection Sort: Example



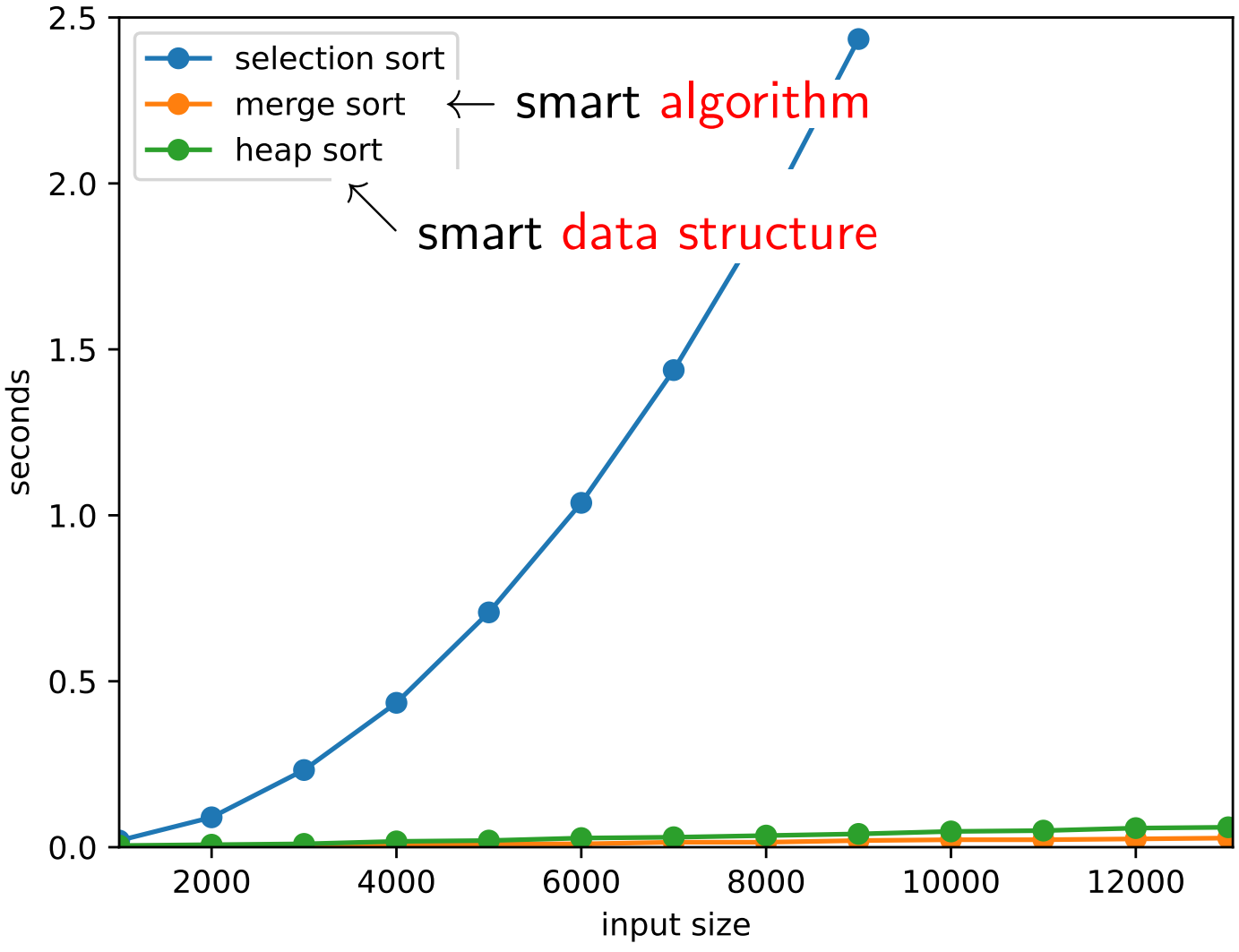
Algorithms for Sorting: Runtime



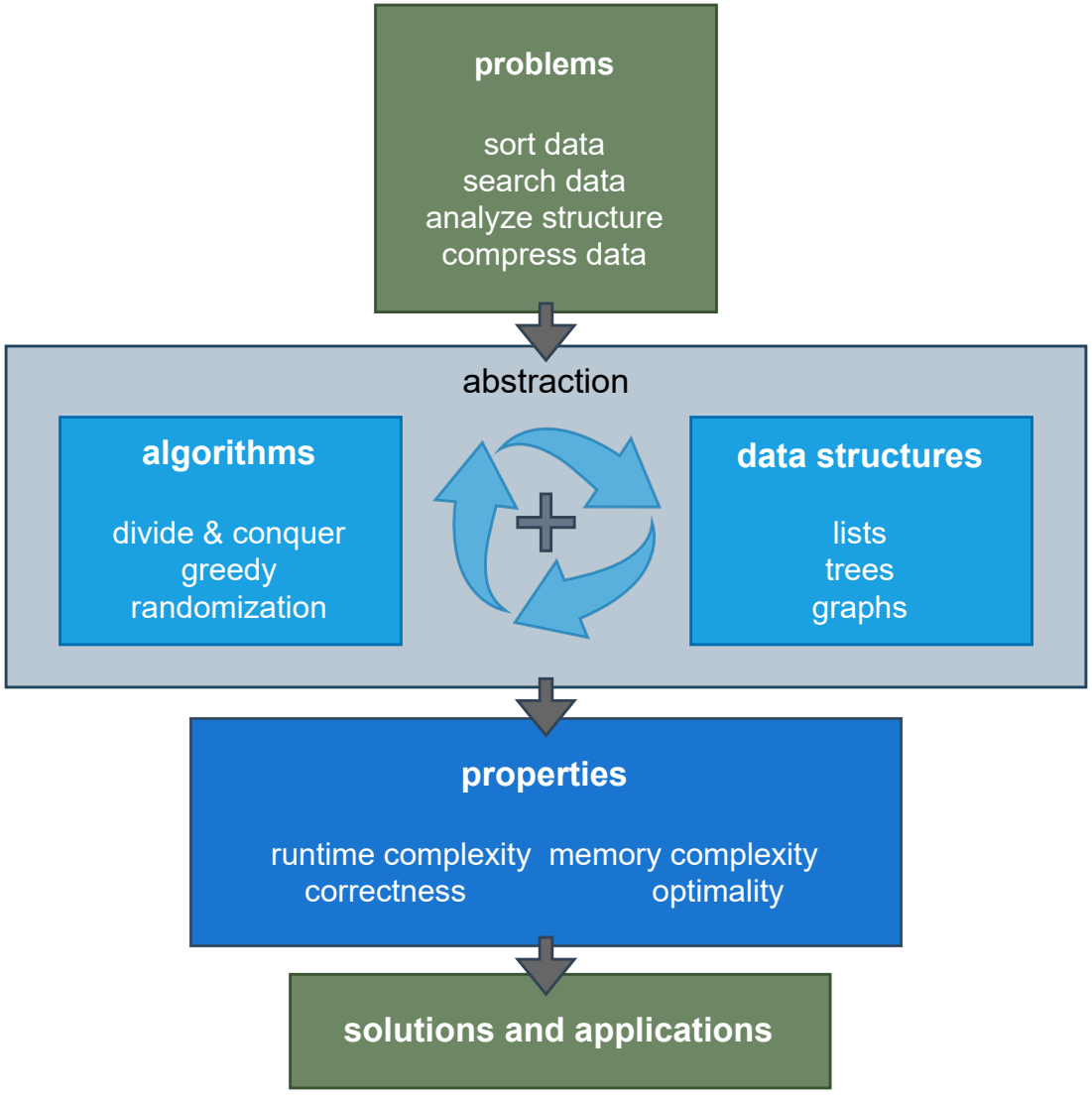
Algorithms for Sorting: Runtime



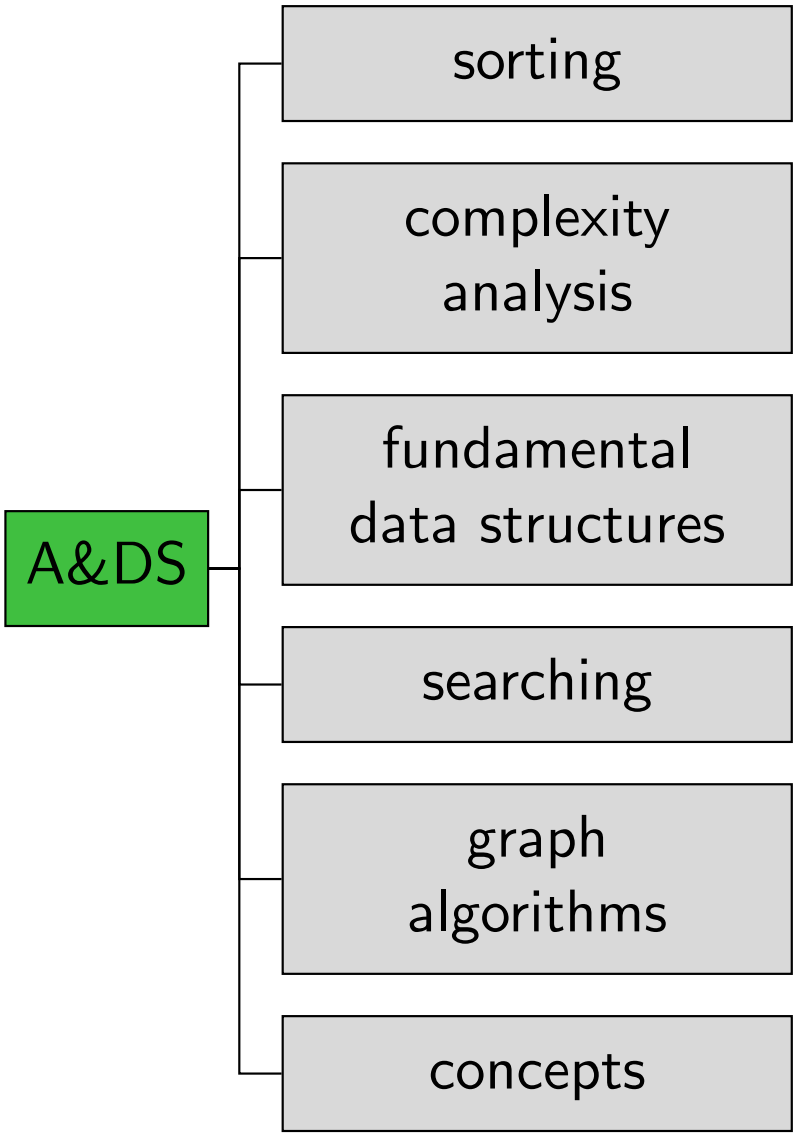
Algorithms for Sorting: Runtime



Algorithms and Data Structures



Content of the Course



Content of the Course