

Theory of Computer Science

A1. Organizational Matters

Gabriele Röger

University of Basel

February 18, 2019

Theory of Computer Science

February 18, 2019 — A1. Organizational Matters

A1.1 Organizational Matters

A1.2 About this Course

A1.1 Organizational Matters

People

Lecturer

Dr. Gabriele Röger

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

People

Tutors

Salomé Eriksson

- ▶ **email:** `salome.eriksson@unibas.ch`
- ▶ **office:** room 04.001, Spiegelgasse 5

Cedric Geissmann

- ▶ **email:** `cedric.geissmann@unibas.ch`
- ▶ **office:** room 04.001, Spiegelgasse 5

Time & Place

Lectures

- ▶ **Monday:** 14:15–16:00
- ▶ **Wednesday:** 16:15–18:00

Room 05.002, Spiegelgasse 5

Time & Place

Exercise Sessions (starting February 25)

- ▶ group 1 (Cedric Geissmann; **in German?**)
 - ▶ **time:** Monday 16:15–18:00
 - ▶ **place:** room 00.003, Spiegelgasse 1
- ▶ group 2 (Salomé Eriksson; in English)
 - ▶ **time:** Monday 16:15–18:00
 - ▶ **place:** room U1.001, Spiegelgasse 1

important: please send me an email with your preferred language until **Wednesday 23:59** (February 20).

Theory Course on the Web

Course Homepage

- ▶ course information
- ▶ slides
- ▶ exercise sheets and additional material

enrolment:

- ▶ <https://services.unibas.ch/>
- ▶ **deadline:** March 18
- ▶ better today, so that you get all relevant emails

Course Material

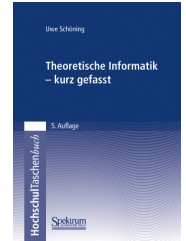
course material:

- ▶ slides (online)
- ▶ textbooks (see next slides)
- ▶ additional material **on request**

Course Material

Textbooks (German)

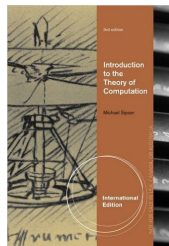
- ▶ Logik für Informatiker
by Uwe Schöning (5th edition)
 - ▶ covers the **part on logic**,
but also advanced topics
beyond the scope of the course
- ▶ Theoretische Informatik – kurz gefasst
by Uwe Schöning (5th edition)
 - ▶ covers **large parts** of the course,
but not the part on logic



Course Material

Textbooks (English)

- ▶ Logic for Computer Scientists
by Uwe Schöning (1st edition)
 - ▶ covers the **part on logic**,
but also advanced topics
beyond the scope of the course
- ▶ Introduction to the Theory of Computation
by Michael Sipser (3rd edition)
 - ▶ covers **large parts** of the course,
but not the part on logic



Target Audience

target audience:

- ▶ B.Sc. Computer Science, 4th semester
- ▶ B.A. Computer Science, 4th or 6th semester as an elective
or if interested in M.Sc. Computer Science degree
- ▶ all other students welcome

prerequisites:

- ▶ basic proof techniques
(mathematical induction, proof by contradiction, ...)
- ▶ basic programming skills

Exam

- ▶ **written exam**, 8 ECTS credits
- ▶ June 12, 14:00–16:00
- ▶ admission to exam: **no prerequisites**
- ▶ must **register** for exam during April 1–15
↔ see <https://philnat.unibas.ch/de/examen/>
- ▶ grade for course determined exclusively by the exam
- ▶ if you fail: **one** repeat attempt in FS 2020

Exercises

Exercise sheets (homework assignments):

- ▶ mostly theoretical exercises
- ▶ some programming exercises

Exercise sessions:

- ▶ live exercises
- ▶ questions about exercise sheets
- ▶ questions about the course
- ▶ participation voluntary but highly recommended

Exercises

- ▶ exercise sheets on course homepage every Wednesday
- ▶ may be solved in **groups of arbitrary size** (recommended: 2–3)
- ▶ due Wednesday the following week
(upload to Courses at <https://courses.cs.unibas.ch/>)
- ▶ scans must be legible (no photos, please)
- ▶ we appreciate \LaTeX submissions

Plagiarism

Plagiarism (Wikipedia)
 is the “wrongful appropriation” and
 and presentation of another author’s “language, thoughts,
 ideas, or creative originality” and the representation of them as
 one’s own original work.

consequences:

- ▶ 0 marks exercise sheet (first time)
- ▶ 0 marks from exam (second time)

in doubt: check with us what is (and isn't) OK **before submitting**

exercises too difficult? we are happy to help!

Questions on Organization



Questions?

A1.2 About this Course

Content: Theoretical foundations of computer science

- A. **background**
 - ▷ mathematical foundations and proof techniques
- B. **logic** (Logik)
 - ▷ How can knowledge be represented?
 - How can reasoning be automated?
- C. **automata theory and formal languages**
(Automatentheorie und formale Sprachen)
 - ▷ What is a computation?
- D. **Turing computability** (Turing-Berechenbarkeit)
 - ▷ What can be computed at all?
- E. **complexity theory** (Komplexitätstheorie)
 - ▷ What can be computed efficiently?
- F. **more computability theory** (mehr Berechenbarkeitstheorie)
 - ▷ Other models of computability

Learning Goals

- ▶ understanding the **capabilities and limitations** of computers
- ▶ working with **formal systems**
 - ▶ comprehending formal **definitions and theorems**
 - ▶ **precise formulation** of definitions, theorems and proofs
 - ▶ analyzing formal problems **precisely**
 - ▶ differentiating statements **within a system** from statements **about a system**

Warning

“Wer’s nicht gewohnt ist,
für den ist es ungewohnt.”
(Prof. Dr. Th. Ottmann)
[If you are not used to it,
it may be unusual for you.]



What can you do?

- ▶ stay on the ball
- ▶ do the exercises
- ▶ pay attention to details
- ▶ ask questions!

Questions about the Course



Questions?