Theory of Computer Science

A1. Organizational Matters

Malte Helmert

University of Basel

February 22, 2016

A1.1 Organizational Matters

People

Lecturer

Prof. Dr. Malte Helmert

- email: `malte.helmert@unibas.ch`
- office: room 06.004, Spiegelgasse 1

People

Assistant

Florian Pommerening

- email: florian.pommerening@unibas.ch
- office: room 04.001, Spiegelgasse 5

People

Tutors

Manuel Heusner, Jendrik Seipp, Silvan Sievers, Salomé Simon

- email: manuel.heusner@unibas.ch
  jendrik.seipp@unibas.ch
  silvan.sievers@unibas.ch
  salome.simon@unibas.ch
- office: room 04.001, Spiegelgasse 5

Time & Place

Lectures

- time: Mon 13:15-15:00, Wed 14:15-16:00
- place: room 05.002, Spiegelgasse 5

Time & Place

Exercise Sessions (starting February 29)

- group 1 (Silvan Sievers)
  - time: Mon 15:15-17:00
  - place: room 05.002, Spiegelgasse 5
- group 2 (Jendrik Seipp)
  - time: Wed 16:15-18:00
  - place: room 05.001, Spiegelgasse 5
- group 3 (Salomé Simon; in English)
  - time: Mon 16:15-18:00
  - place: room 05.001, Spiegelgasse 5

important: please choose your preferences in Courses
(https://courses.cs.unibas.ch/)
until Wednesday 23:59 (February 24)

Theory Course on the Web

Course Homepage

`http://informatik.unibas.ch/fs2016/theorie-der-informatik/`

- course information
- slides
- exercise sheets and additional material

registration:

- `https://services.unibas.ch/`

Course Material

course material:

- slides (online + printed handouts)
- textbooks (see next slides)
- additional material on request

Course Material

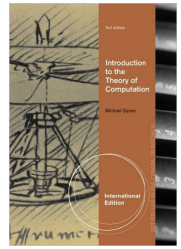
Textbooks (German)

- 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:

- Bachelor Informatik (computer science), 2nd semester
- other students welcome

prerequisites:

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

Exam

- written exam
- 6 ECTS credits
- admission to exam: 50% of the exercise marks
- grade for course determined exclusively by the exam
- July 1, 10:00-12:00 (room TBA)
- no repeat exam

Exercises

Exercise sheets (homework assignments):

- mostly theoretical exercises
- some programming exercises

Exercise sessions:

- live exercises (not relevant for exam admission)
- questions about exercise sheets
- questions about the course
- participation voluntary but highly recommended

Exercises

- exercise sheets on course homepage every Monday
  (exception: first exercise sheet available Wed, February 24)
- solved in groups of two (note: $2 < 3$)
- due Wednesday following week
  (pigeon holes Spiegelgasse 1 or upload to Courses)
- scans must be legible (no photos, please)
- one bonus point per sheet for LaTeX submissions

Plagiarism

Plagiarism (Wikipedia)

> *Plagiarism is the "wrongful appropriation" and "stealing and publication" of another author's "language, thoughts, ideas, or expressions" and the representation of them as one's own original work.*

consequences:

- 0 marks for the exercise sheet (first time)
- exclusion from exam (second time)

if 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

Contents

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. computability theory (Berechenbarkeitstheorie)
   ▷ What can be computed at all?

E. complexity theory (Komplexitätstheorie)
   ▷ What can be computed efficiently?

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?