Algorithms

Some information on the course (as of April 7, 2022)

Lecture: Friday, 11:00 **(sharp!)**, Humboldt-Hörsaal.

Instructor: Prof. Martin **Dietzfelbinger**, Zusebau 1045, Tel. (03677)69-2656,
e-mail: `martin.dietzfelbinger@tu-ilmenau.de`
Office hour: Thursdays, 17:00–18:00 or by appointment.

Discussion group (obligatory, starting in week 17):
Monday (odd weeks), 15:00 – 16:30, Sr K 2026
Tuesday (odd weeks), 13:00 – 14:30, Sr H 1527

Instructor for discussion group: Arindam Biswas,
e-mail: `arindam.biswas@tu-ilmenau.de`
Office hour: by appointment

For communication, we'll use the Moodle2 system:
`https://moodle2.tu-ilmenau.de/course/view.php?id=4074`
Switch languages with globe symbol. Login with your university login and password.
→ Fakultät IA → Research in Computer and Systems Engineering → Algorithms

Please register for this course as soon as possible and thus make sure you receive the messages coming through the system.

Textbook: "Algorithms" by S. Dasgupta, C. H. Papadimitriou, and U. V. Vazirani. McGraw-Hill, 2006.
University Library: 79: Lehrbuchsammlung; Signature: INF ST 134 D229
Possibly a preliminary pdf version can be found on the web (which may contain more errors than the actual book).

It is essential you get hold of the book *immediately*. There will be some extra notes by the instructor (Moodle2); most of the material is described in the book, page numbers will be given. The instructor will use the blackboard a lot. Beware: Some of the material on the blackboard (and in the notes) is **different from** what is in the book or **not covered** in the book.

So please take notes.

Videos of lectures in the winter semester will be provided. It is possible to follow the course remotely.

Working style: We expect that you study the material covered in class **on your own** again, before the next class, and you prepare solutions for problems for the discussion groups.

Hint: Form study groups, work at fixed hours. If things are hard to understand, ask one of the instructors (after class, in the office hour). Ask early, not one week before the exam.

In the course the students get a general understanding about what *algorithms* are, when we consider them as efficient and when we do not. In the center of attention are about 30 particularly clever and fundamental algorithms, falling into several categories. They are explored according to the following criteria:

- What is the computational problem solved by the algorithm?
- What is the input, what is the output?
- How does the algorithm work? (Pseudocode — run on example input.)
- In some cases, one has to **prove** that the algorithm is **correct**.
- In some cases, one has to provide suitable data structures that help speeding up the algorithm.
- The running time of the algorithm is described in dependence on the size of the input.
- In some cases, the time bound needs a **proof**.
- Can we do better?

Towards the end, we discuss the notion of NP-completeness. NP-complete problems are conjectured not to have efficient algorithms.

Exam

The exam for this course is in written form, as a 90-minute closed books exam, in July or August. You will be given more information later. But it is always good to understand the algorithms, be able to explain them and to work them on example inputs.

Topics

- Introduction, big-O notation
- Integer arithmetic
- Modular arithmetic
- Primality testing and generating prime numbers
- The RSA cryptosystem
- Divide-and-conquer algorithms
- Integer multiplication made faster (Karatsuba)
- Fast Fourier Transform
- Representing Graphs
- Exploring graphs: Depth-first search
- Cycles in graphs, acyclic graphs
- Short paths in graphs: Breadth-first search and Dijkstra's algorithm
- Greedy algorithms for graph problems: Minimum spanning trees
- Huffman coding
- Dynamic programming: Shortest paths again, edit distance
- NP-completeness