Level: bachelor

Course title: Analysis of Algorithms (I152)

Status: obligatory

ECTS: 8

Requirements: Discrete Mathematics 1

Learning objectives

The goal of the course is to present an introduction to basic concepts of computability theory i.e. to mathematical models of the intuitive notion of an algorithm and the fundamental techniques for estimating computational complexity of algorithmic problems.

Learning outcomes

Minimal: Understanding the necessity and the basic idea behind the mathematical formalisation of the notion of an algorithmic problem; designing simplest (non-branching) programs of Turing machines and developing the capability of analysing the time complexity of classical algorithms by solving the well-known computational problems.

Desirable: Successful students should be able to design – by solving advanced problems concerning recursive functions and sets – more involved (branching) programs of Turing machines, and to apply theoretical knowledge in the area of computational complexity to analysis of problems and corresponding programs that are met in everyday programming practice.

Syllabus

Theoretical instruction

The intuitive notion of an algorithm and its formalisation, Church thesis. The basics of the theory of recursive functions. Recursive and recursively enumerable sets and languages. Turing machines and related computational models. Complexity of machines and complexity classes. An overview of some classical problems of polynomial time complexity. Non-deterministic Turing machines and their complexity. Reductions and completeness. **NP**-complete problems.

Practical instruction

Primitive recursive functions. Applications of theorems on sums and products. The minimisation operator. Applications of the majorisation theorem. Recursive sets. Computing functions on Turing machines in the {|,*}-system. Composition of machines. Routines for the iterated sum, product, function composition, the primitive recursion scheme, minimisation. Representations of graphs. Basic algorithms on graphs. Mutual reductions of various problems from mathematical logic, combinatorics, discrete optimisation.

Weekly teaching load				Other: 0
Lectures: 3	Exercises: 3	Other forms of teaching: 0	Student research: 0	