Study programme(s): Computer Science

Level: bachelor

Course title: Discrete Structures 2

Lecturer: Dragan Mašulović, Maja Pech

Status: obligatory

ECTS: 6

Requirements: Discrete Structures 1

Learning objectives

In this course students shall acquire deeper knowledge of discrete processes that are vital to computer science and will understand the notions such as universal and existential quantification; recursive mathematical definitions; fundamental counting techniques; classical algebraic structures and applications in coding theory.

Learning outcomes

At the end of the course a successful student will be able to perform basic calculations in the predicate logic, be able to produce and understand recursive mathematical definitions, solve elementary counting problems, understand basic facts about classical algebraic structures and apply this knowledge to basic coding techniques.

Syllabus

- Predicate logic (Universal and existential quantification)
- Structural induction
- Recursive mathematical definitions
- Limitations of predicate logic (e.g., expressiveness issues)
- Basic Counting
- The pigeonhole principle
- Permutations and combinations
- Inclusion-Exclusion
- Solving recurrence relations
- Basic modular arithmetic
- Concrete algebraic structures (permutations as groups; integers and matrices as rings; rational, real and complex numbers as fields; finite fields)
- Introduction to coding theory

Literature

D. J. Hunter: "Essentials of Discrete Mathematics", Jones and Bartlett Learning, 2017
J. Matoušek, J. Nešetril: "Invitation to Discrete Mathematics", Oxford University Press, 2008
S. G. Krantz: "Discrete Mathematics Demystified", McGraw-Hill, 2009

Weekly teaching load				
Lectures: 3	Exercises: 2	Practical Exercises: 0	Student research: 0	Other: **0**

Teaching methodology

Blackboard lectures, Blackboard exercises

Grading method (maximal number of points 100)

Pre-exam obligations	**points**	**Final exam**	**points**
Colloquium 1	**30**	*Oral exam*	*40*
Colloquium 2	**30**		