



*Course syllabus*

# Diskreta strukturer Discrete Structures

**EDAA75, 7,5 credits, G1 (First Cycle)**

**Valid for:** 2023/24

**Faculty:** Faculty of Engineering, LTH

**Decided by:** PLED C/D

**Date of Decision:** 2023-04-18

## General Information

**Main field:** Technology.

**Compulsory for:** C1

**Elective Compulsory for:** D1

**Language of instruction:** The course will be given in English

## Aim

The course is intended to introduce some of the basic formal concepts and terminology pervading all areas of computer science, and to establish a common lexicon, including notational conventions and nomenclature, that subsequent courses can build upon. This includes an introduction to abstract set theory, relations, functions, and ordered sets, Boolean algebra, logic and proof techniques, as well as structures such as graphs and trees. Furthermore, the course discusses basic algorithms on graphs, an introduction to combinatorics, some fundamental proof strategies, and basic order structures such as lattices and complete partial orders (CPOs).

## Learning outcomes

*Knowledge and understanding*

For a passing grade the student must

- demonstrate an understanding of basic notions of set theory, such as equivalence, cardinality, countability, infinite sets,
- be able to characterize functions, injective/surjective/bijective functions, relations, partial and total orders and their properties, equivalence relations,
- understand basic proof techniques such as induction,
- be familiar with Boolean algebra and first order logic,

- understand fundamental structures such as trees and graphs,
- have an understanding on foundations of constructing algorithms on graphs
- know basic concepts of combinatorics (e.g. permutations, combinations),
- know about basic types of ordered sets, such as lattices and complete partial orders,
- be able to apply some fundamental proof strategies, such as direct proofs, contrapositive proofs, proofs by contradiction.

#### *Competences and skills*

For a passing grade the student must

- be able to use notation associated with sets, relations, functions, and orders to define structures and discuss their properties,
- be able to use induction to prove properties of infinite sets of objects,
- be able to manipulate, transform and simplify Boolean terms according to the laws of Boolean algebra,
- be able to work with trees and graphs, such as devising proofs of properties,
- be able to implement simple algorithms and tests for properties on discrete structures,
- be familiar with techniques for building algorithms on graphs and some basic examples,
- be able to work with permutations and combinations and use them to in computational problems,
- be able to use and apply appropriate types of order structures such as lattices and CPOs,
- be able to use basic proof strategies to construct simple proofs.

#### *Judgement and approach*

For a passing grade the student must

- be able to apply sets, graphs, and trees to represent aspects of real-world problems and build algorithms on those structures,
- show the ability to devise an appropriate proof strategy and technique for a given problem.

## Contents

Sets, set equivalence, infinite sets, countability, functions, properties of functions (injective, surjective, bijective functions), relations, orders (total and partial), transitivity, (anti) symmetry, reflexivity, equivalence relations and classes, complete partial orders, Boolean algebra, predicate logic, proofs, induction, graphs, trees, graph algorithms, combinatorics, order structures, proof strategies.

## Examination details

**Grading scale:** TH - (U,3,4,5) - (Fail, Three, Four, Five)

**Assessment:** Written exam. For a passing grade of the course the compulsory course items must also be completed. The final grade of the course is based on the result of the written exam.

The examiner, in consultation with Disability Support Services, may deviate from the regular form of examination in order to provide a permanently disabled student with a form of examination equivalent to that of a student without a disability.

#### **Parts**

**Code:** 0121. **Name:** Compulsory Course Items.

**Credits:** 3. **Grading scale:** UG. **Assessment:** Approved mandatory elements

**Code:** 0221. **Name:** Written Examination.

**Credits:** 4,5. **Grading scale:** TH. **Assessment:** The final grade of the course is based on the result of the

written examination. In order to pass, the compulsory course items must also be completed. **Contents:**  
Written examination.

## Admission

### Admission requirements:

- EDAA20 Programming and Databases or EDAA45 Introduction to Programming or EDAA50 Programming, First Course or EDAA55 Programming, First Course or EDAA65 Programming, First Course

**The number of participants is limited to:** No

**The course overlaps following course/s:** EDAF10, EDAA40

## Reading list

- David Makinson: Sets, Logic, and Maths for Computing, 3rd ed. Springer. The book is also available as a free PDF/eBook download within the LTH network: <https://link.springer.com/book/10.1007/978-3-030-42218-9>.

## Contact and other information

**Course coordinator:** Jacek Malec, [jacek.malec@cs.lth.se](mailto:jacek.malec@cs.lth.se)

**Course homepage:** <http://cs.lth.se/edaa40>