



ID1020 Algorithms and Data Structures 7.5 credits

Algoritmer och datastrukturer

This is a translation of the Swedish, legally binding, course syllabus.

Establishment

Course syllabus for ID1020 valid from Autumn 2019

Grading scale

A, B, C, D, E, FX, F

Education cycle

First cycle

Main field of study

Technology

Specific prerequisites

- Completed course ID1018 Programming I, or the equivalent.
- Completed course IS1200 Computer Hardware Engineering, or the equivalent.

Language of instruction

The language of instruction is specified in the course offering information in the course catalogue.

Intended learning outcomes

The aim of the course is to provide a solid knowledge on how to design and analyse the most important classes of algorithms. The course intends to give the students skills that give them the opportunity to design computer program that use time and memory in an efficient way independently. The students should learn to identify problems that are unreasonably demanding in terms of resources or that indeed are impossible to solve with common computing models. At the end of the course the students should be able to develop their own algorithms to solve problems and be able to compare different solutions and how well they function.

Upon completion of the course, the students should be able to

- develop and implement algorithms and data structures and analyse them regarding correctness.
- define the terms P, NP, NP-completeness and computability.
- analyse how efficient algorithms and data structures are based on different measures on efficiency such as time complexity and memory complexity.
- write programs that use algorithms and data structures with the help of good programming principles such as the specification of APIs and the use of tests that utilise algorithms or data structures.
- solve problems through the use of data structures such as linear lists, stacks, queues, hash tables, binary trees, heaps, binary search trees, and graphs, and write programs for the solutions.
- solve problems by using algorithm design methods such as greedy algorithms, decomposition, dynamic programming, backtracking, and write programs for the solutions.
- given a specific problem, either design an appropriate data structure or create an algorithm that solves the problem or identify the problem as one that cannot be solved efficiently.

Course contents

Basic analysis

- Asymptotic analysis of complexity borders in the worst case and on average.
- To identify differences between behaviours of algorithms in the best case, the worst case, and on average.
- The Master theorem
- The notations of ordo, omega and theta.
- The most common complexity classes. Empirical estimate of complexity. Complexity in time and memory.
- the use of recursion to analyse algorithms.

Strategies for algorithms and data abstraction.

- Brute-force algorithms.
- Greedy algorithms.

- Decomposition algorithms.
- Backtracking.
- Heuristics.

Basic algorithms

- Simple numerical algorithms.
- Sequential and binary search algorithms.
- Quadratic sorting algorithms (selection, insertion).
- Sorting in $O(N \log N)$ (Quicksort, Heapsort, Mergesort).
- Hash tables including such that avoid collision.
- Binary search trees.
- Representations of graph (neighbour lists, neighbour matrices).
- Depth first search and Width first search.
- Shortest path algorithms (the algorithms of Dijkstra and Floyd).
- Transitive closure (Floyd's algorithm).
- Minimum spanning trees (the algorithms of Prim and Kruskal).
- Topological sorting.

Basic computability

- Solvable and unsolvable problems.
- computable and non-computable functions.
- The halting problem
- Consequences of undecidability.

P and NP.

- Definitions of the classes P and NP.
- NP-completeness (Cook's theorem).
- Some common NP-complete problems.

Course literature

Algorithms, 4th Edition, 2011. Robert Sedgewick and Kevin Wayne. Addison-Wesley.
ISBN-10: 032157351X.

Examination

- ARBA - Course work, 4.5 credits, grading scale: P, F
- TENA - Written exam, 3.0 credits, grading scale: A, B, C, D, E, FX, F

Based on recommendation from KTH's coordinator for disabilities, the examiner will decide how to adapt an examination for students with documented disability.

The examiner may apply another examination format when re-examining individual students.

If the course is discontinued, students may request to be examined during the following two academic years.

The expected learning outcomes that have been defined above are examined both through continuous examination and through a written examination. The entire work of the student is performed individually. There are no group projects.

In agreement with KTH's coordinator for disabilities, it is the examiner who decides to adapt the examination for students in possession of a valid medical certificate. The examiner may permit other examination forms at the re-examination of few students

Ethical approach

- All members of a group are responsible for the group's work.
- In any assessment, every student shall honestly disclose any help received and sources used.
- In an oral assessment, every student shall be able to present and answer questions about the entire assignment and solution.