



HÖGSKOLAN I GÄVLE

Algorithms and Datastructures 7.5 cr

Algoritmer och datastrukturer 7,5 hp

Set by Faculty of Engineering and Sustainable Development

Version

Set at

Valid from

2/26/18

HT2018

Level	G1F
Education level	First cycle
Course identifier	DVG329
Credits	7.5 cr
Main field of study	Computer Science
Subject group	Computer Technology
Disciplinary domain	Technology 100.0 %

Learning outcomes

After completion of the course the student shall be able to

1. explain and use both fundamental terminology and programming techniques within the areas of algorithms and data structures, e.g. algorithms, abstract data types, interfaces, documentation, implementation, test, traversal and recursion
2. explain and use the Big-O notation for comparative analysis and suitable selections of algorithms and their implementations for specific problems
3. explain, describe, use and implement (in a high-level programming language) sequential data structures, e.g. lists, queues, stacks and priority queues with accompanying algorithms
4. explain, describe, use and implement (in a high-level programming language) tree data structures, e.g. binary trees, heaps and search trees with accompanying algorithms
5. explain, describe, use and implement (in a high-level programming language) various common data structures and search algorithms that follow standard Big-O functions
6. explain, describe, use and implement (in a high-level programming language) common search algorithms that follow standard Big-O functions.

Course content

The term "abstract data type" (ADT) and "algorithm"

The relationship between an ADT's interface and it's implementation

The definition and implementation of ADTs such as lists, stacks, queues, binary trees and hash tables

Various algorithms for linear, binary and hash searches
 Algorithms for sorting (e.g. insertion, selection, bubble, merge and quick sort) as well as traversal of data structures
 Introduction to algorithm analysis and the Big-O notation
 Implementation of an algorithm using iteration and recursion
 The relationship between ADTs and their implementations in a high-level programming language using data types, classes, modules and interfaces
 Documentation and test of code and data types

Teaching Lectures, practical laboratories, a supervised project, written reports and seminar presentations

Prerequisites Object-Oriented Design and Programming 7.5 credits and Calculus 7.5 credits or equivalent

Examination Written examination, written reports and seminars.

0010 Written examination examines learning outcomes 1-6, grades A-F.
 0020 Laboratory work examines learning outcomes 1-6, grades Pass or Fail.
 0030 Programming project examines learning outcomes 1, 3, 4-6, grades Pass or Fail.

Grade A, B, C, D, E, Fx, F

Other regulations Degree Criteria for final grade will be given by course responsible or examiner latest at the beginning of the course.

Sustainable environment A minor part of the course content deals with sustainable development.

Module

0010	Written Examination: Theory and nomenclature	3 cr	Grade: AF
0020	Practical Laboratories: Usage and Implementation	2 cr	Grade: UG
0030	Programming Project	2.5 cr	Grade: UG