



## Course syllabus

Faculty of Technology  
Department of Computer Science

1DV507 Programmering och datastrukturer, 7,5 högskolepoäng  
Programming and Data Structures, 7.5 credits

**Main field of study**  
Computer Science

**Subject Group**  
Informatics/Computer and Systems Sciences

**Level of classification**  
First Level

**Progression**  
G1F

**Date of Ratification**  
Approved by Faculty of Technology 2015-05-22  
The course syllabus is valid from spring semester 2016

**Prerequisites**  
This is a follow-up course that requires an introductory course in Java (7.5 credits).

## Objectives

After the course the student should be able to:

- briefly describe the object-oriented software development process
- describe what a design pattern is and why they are used
- describe the concepts algorithm, data structure, and time complexity
- describe and implement a few simple data structures like list, stack, queue, hash table, binary tree, and binary heap
- describe different types of testing and be able to perform unit testing using JUnit
- describe the most commonly used data structures in the Java Standard Library
- describe the object-oriented concepts inheritance and polymorphism
- model and implement systems using the object-oriented concepts inheritance and polymorphism

## Content

The following theoretical parts are treated:

- graphical user interfaces
- recursion
- introduction to algorithms and data structures
- simple search and sorting algorithms
- introduction to time complexity
- implementations of simple data structures like list, stack, queue, hash table, binary

- tree, and binary heap
- testing in general and unit testing in detail
- inheritance and polymorphism in object oriented programming
- object oriented modelling using inheritance

The following parts in Java are treated:

- exceptions
- recursive method calls
- data structures in the Java Standard Library
- JUnit testing
- interfaces, inheritance, and polymorphism
- generic classes

## Type of Instruction

Teaching consists of lectures and practical assignments. Practical assignments are individual or carried out in groups. The course can also be given as a distance learning course.

## Examination

The course is assessed with the grades A, B, C, D, E, Fx or F.

The grade A constitutes the highest grade on the scale and the remaining grades follow in descending order where the grade E is the lowest grade on the scale that will result in a pass. The grade F means that the student's performance is assessed as fail (i.e. received the grade F).

Assessment of the students's performance is made through written and/or oral examination or practical programming examination and/or presentation of mandatory assignments. The assessment method is decided at the start of the course.

Students who do not pass the regular examination are given the opportunity to do a resit examination shortly after the regular examination.

## Course Evaluation

During the course or in close connection to the course, a course evaluation is to be carried out. The result and analysis of the course evaluation are to be communicated to the students who have taken the course and to the students who are to participate in the course the next time it is offered. The course evaluation is carried out anonymously. The compiled report will be filed at the Faculty.

## Credit Overlap

This course cannot be part of a degree in combination with another course in which the content fully or partly correspond to the content of this course: 1DV007 Programming and Data Structures, 7.5 credits

## Other

Grade criteria for the A–F scale are communicated to the student through a special document. The student is to be informed about the grade criteria for the course by the start of the course at the latest.

## Required Reading and Additional Study Material

### Required reading

Horstmann, Cay S., *Big Java: Late Objects*, John Wiley & Sons, 2012 or later. Pages 440 (900).

FTK, *Distributed material*. Pages 100.