Object-Oriented Programming

Programme course

6 credits

Objektorienterad programmering

TND002

Valid from:

Determined by
Board of Studies for Computer Science
and Media Technology

Date determined
2017-01-25

Main field of study

Computer Science and Engineering

Course level

First cycle

Advancement level

G1X

Course offered for

- Media Technology and Engineering, M Sc in Engineering
- Communication and Transportation Engineering, M Sc in Engineering

Entry requirements

Note: Admission requirements for non-programme students usually also include admission requirements for the programme and threshold requirements for progression within the programme, or corresponding.

Prerequisites

Introductory course in programming

Intended learning outcomes

The course should give knowledge of object-oriented program development and programming in an object-oriented language, Java. After this course the student should be able to:

- write simple programs with input and output of numbers and text, using the command prompt as well as dialog boxes.
- explain the difference between built-in data types and class types, and the difference between simple variables and reference variables.
- explain the difference between instance variables and class variables, and the difference between instance methods and class methods.
- design and implement classes, and use them.
- use common standard classes from the Java API.
- write programs for file processing, in particular text files.
- explain the concepts inheritance, polymorphism and dynamic binding, and the concepts abstract class and interface.
- design and implement class hierarchies, and use them.
- write programs with simple graphical user interfaces.
- explain the concept event-driven program.
- for a given problem, develop an object-oriented program consisting of a number of classes, and document this in UML.

Course content

The Java programming language, variables and data types, input and output, control structures.
Classes and objects, instance variables and instance methods, class variables and class methods.
Object-oriented programming, class diagrams and UML.
Standard classes and the Java API,
Inheritance, subclasses and superclasses, polymorphism and dynamic binding, abstract classes and interfaces.
Streams and files, exceptions.
Programs with graphical user interfaces, graphical components and layout managers, event-driven programs.

Teaching and working methods

Lectures, lessons, and laboratory work.

Examination

LAB1	Laboratory work	3 credits	U, G
DAT1	Computer examination	3 credits	U, 3, 4, 5

Grades

Four-grade scale, LiU, U, 3, 4, 5

Other information

Supplementary courses: Programming in C++

Department

Institutionen för teknik och naturvetenskap

Director of Studies or equivalent

Camilla Forsell

Examiner

Mark E Dieckmann

Course website and other links

http://www2.itn.liu.se/utbildning/kurs/index.html?coursecode=TND002

Education components

Preliminary scheduled hours: 52 h
Recommended self-study hours: 108 h

Course literature

Java direkt med Swing, Jan Skansholm, Studentlitteratur, 7:e upplagan år 2013 (eller senare).

Common rules

Regulations (apply to LiU in its entirety)

The university is a government agency whose operations are regulated by legislation and ordinances, which include the Higher Education Act and the Higher Education Ordinance. In addition to legislation and ordinances, operations are subject to several policy documents. The Linköping University rule book collects currently valid decisions of a regulatory nature taken by the university board, the vice-chancellor and faculty/department boards.

LiU's rule book for education at first-cycle and second-cycle levels is available at http://styrdokument.liu.se/Regelsamling/Innehall/Utbildning_pa_grund-_och_avancerad_niva.