DD1385 Software Engineering 6.0 credits

Programutvecklingsteknik

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

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

Establishment

On 04/21/2020, the Head of the EECS School has decided to establish this official course syllabus to apply from the autumn semester 2020, registration number J-2020-0582.

Grading scale

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

Education cycle

First cycle

Main field of study

Technology

Specific prerequisites

Completed course in programming corresponding to DD1310/DD1311/DD1312/DD1314/DD1315/DD1316/DD1318/DD1331/DD1337/DD100N/ID1018 an

Active participation in a course offering where the final examination is not yet reported in LADOK is considered equivalent to completion of the course. This applies only to students

who are first-time registered for the prerequisite course offering or have both that and the applied-for course offering in their individual study plan.

Language of instruction

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

Intended learning outcomes

After completion of the course, the student should be able to:

- account for object-oriented concepts and technologies as loose coupling, encapsulation, inheritance and polymorphism, and use them in your own programming,
- describe and apply criteria for good object-oriented design
- recognise and explain common design patterns in object-oriented software development as well as choose appropriate patterns for simple applications
- draw UML class diagrams that in a clear and legible way document your own programming work,
- interpret and explain UML class diagrams,
- use advanced language elements in Java, such as exceptions, threads, graphical components, object collections and web programming in your own programming work,
- explain the structure and function of the elements of the language,
- explain and apply test driven software development

in order to:

- be prepared to participate in software development projects,
- be able to work with software development in Java.

Course contents

Object-Oriented Programming in Java. Use of class libraries in Java for object collections, simple graphical components, advanced graphical components, client-server programming and threads. UML class diagrams.

Design patterns for object-oriented software development.

Criteria for good design. Testing and test driven software development.

Orientation in object-oriented analysis.

Orientation in common development methodologies.

Laboratory sessions in Java that all are documented with UML class diagrams.

Examination

- LAB1 - Laboratory assignments, 3.0 credits, grading scale: A, B, C, D, E, FX, F
- PRO1 - Project, 1.5 credits, grading scale: A, B, C, D, E, FX, F
- TEN1 - Examination, 1.5 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.

Other requirements for final grade

Passed laboratory assignments, project and examination.

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.