DD1331 Fundamentals of Programming 5.0 credits

Grundläggande programmering

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

Establishment

Grading scale

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

Education cycle

First cycle

Main field of study

Technology

Specific prerequisites

Language of instruction

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

Intended learning outcomes

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

- describe the parts of and the terminology for a computer system at a general level, such as CPU, memory, operating system and user interface,
- pull, push, update and document changes in code with a version control system,
- describe the source code for a computer program with correct terminology,
- describe and apply basic data types, classes and type conversions,
- describe, apply and debug flow control as well as logical and arithmetic expressions,
- describe a variable's scope and extent,
- graphically describe the connection between variable names, types and data
- debug short programs written by others,
- debug own longer programs,
- write a specification for a longer program,
- use and evaluate good programming practices,
- implement, debug, and with correct terminology describe recursive algorithms,
- combine built in functions in the programming language with own functions to solve programming problems,
- choose the right data structure to solve programming problems so that the solution scales well,
- calculate and justify the time complexity for simple algorithms

to be able to

- perform calculations and solve programming problems
- use KTH's computer system
- be prepared for the next course in computer science.

For higher grades, the student should also be able to

- implement an interactive graphical user interface.

Course contents

- Basic computer science concepts
- The version control system Git
- Basic programming in Python
- Abstract data types
- Classes
- Recursion, hierarchical decomposition
- Classical data structures: lists, stacks, queues, hash tables
- Classical algorithms for search and sorting
- Introduction to algorithm analysis

Several smaller programming assignments as well as a larger individual programming assignment with high requirements of structuring and specification of included modules.

Course literature

Will be announced 10 weeks before the start of the course on the course web.

Examination

- TEN2 - Examination, 1.0 credits, grading scale: P, F
- LAB2 - Laboratory Assignment, 2.0 credits, grading scale: A, B, C, D, E, FX, F
- LAB1 - Laboratory Assignment, 2.0 credits, grading scale: P, 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.

If there are special needs, the examination can be carried out in a different way.

LAB1 consists of theoretical assignments. LAB2 is an individual project.

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.