



LUNDS UNIVERSITET
Lunds Tekniska Högskola

Course syllabus

Programmeringsteknik, grundkurs Introduction to Programming

EDAA85, 9 credits, G1 (First Cycle)

Valid for: 2023/24

Faculty: Faculty of Engineering, LTH

Decided by: PLED C/D

Date of Decision: 2023-04-18

General Information

Main field: Technology.

Compulsory for: E1, F1, I1, M1, Pi1

Language of instruction: The course will be given in Swedish

Aim

The goal of the course is to give students an introduction to programming, especially as a way to solve problems from different application areas. The focus is on programming skills with problem solving and step-by-step development.

Learning outcomes

Knowledge and understanding

For a passing grade the student must

- explain basic concepts in imperative and object-oriented programming
- explain and give examples of the use of basic data types and simple algorithms
- choose basic data types appropriate for solving given problems
- interpret (well-written) program code to correctly describe what happens when it is executed
- show understanding of the computer's usefulness as a calculation tool

Competences and skills

For a passing grade the student must

- independently implement programs to solve problems in different application domains
- independently create algorithms that use basic data types and collections (lists, sets and key-value tables)

- construct program code, given a simpler algorithm formulated in human language
- construct and structure programs consisting of several classes, methods and functions
- incrementally develop, test and debug programs
- use numpy for basic simulations and engineering calculations

Judgement and approach

For a passing grade the student must

- assess which basic data types and algorithms are suitable for solving different problems
- evaluate existing code to find and fix simple programming errors

Contents

- Basic program constructs such as functions, iteration, and conditional statements.
- Basic values and types such as integers, floats, booleans, and strings.
- Variables and assignment.
- Basic data types for composite values, such as lists, tuples, sets, and key-value tables.
- Input, printing and files.
- Simple algorithms for searching, sorting and the like.
- Using existing libraries and classes.
- Basic knowledge of object-oriented programming, classes and methods.
- Orientation on inheritance.
- Basic execution model with function calls, parameter passing, objects and method calls.
- Basic programming methodology with step-by-step development, testing and debugging.
- Basic use of programming tools.
- Introduction to computational programming.
- Python 3 is used as the programming language.

Examination details

Grading scale: TH - (U,3,4,5) - (Fail, Three, Four, Five)

Assessment: For final grades, passed assignments in computational programming, passed programming assignments and a passed exam are required. In order to participate in the exam, passed programming assignments are required. The final grade for the course is based on the grade of the exam.

The examiner, in consultation with Disability Support Services, may deviate from the regular form of examination in order to provide a permanently disabled student with a form of examination equivalent to that of a student without a disability.

Parts

Code: 0123. **Name:** Exam.

Credits: 3. **Grading scale:** TH. **Assessment:** Approved examination. **Contents:** Written digital examination.

Code: 0223. **Name:** Programming Assignments.

Credits: 4,5. **Grading scale:** UG. **Assessment:** Approved programming assignments. **Contents:** Programming assignments. **Further information:** Detailed regulations regarding the completion of programming tasks will be available on the learning platform.

Code: 0323. **Name:** Computational Programming.

Credits: 1,5. **Grading scale:** UG. **Assessment:** Approved assignments in computational programming **Contents:** Assignments in computational programming

Admission

The number of participants is limited to: No

The course overlaps following course/s: EDA010, EDA011, EDA015, EDA016, EDA017, EDA390, EDA500, EDA501, EDA616, EDA618, EDAA10, EDAA20, EDAA50, EDAA55, EDAA65, EDAA45, EDAA70, EDAA80, EDAA90

Reading list

- Allen B. Downey: Think Python, How to Think Like a Computer Scientist. O'Reilly, 2015, ISBN: 9781491939369.

Contact and other information

Course coordinator: Patrik Persson, patrik.persson@cs.lth.se

Course coordinator: Sandra Nilsson, sandra.nilsson@cs.lth.se

Course homepage: <https://cs.lth.se/edaa85/>