DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DIT441 Introduction to Functional Programming, 7.5 credits

Introduktion till funktionell programmering, 7,5 högskolepoäng

First Cycle

Confirmation

This course syllabus was confirmed by Department of Computer Science and Engineering on 2021-11-15 to be valid from 2022-08-29, autumn semester of 2022.

Field of education: Science 100%
Department: Department of Computer Science and Engineering

Position in the educational system

The course is compulsory within the Computer Science, Bachelor´s Programme. It is also a single subject course at the University of Gothenburg.

The course can be part of the following programme: 1) Computer Science, Bachelor's Programme (N1COS)

Main field of studies	*Specialization*
Computer Science	G1N, First cycle, has only upper-secondary level entry requirements

Entry requirements

General entrance requirements for university studies and the Swedish upper secondary course Mathematics D or Mathematics 3 c or equivalent.

Learning outcomes

After completing the course the student is expected to be able to:

Knowledge and understanding
- describe the basic concepts of modern functional programming languages, such as: data types, first- and higher-order functions, lazy evaluation, infinite data structures

- describe a basic repertoire of functional programming techniques, such as: recursion, testing, the role of data types in modelling and problem solving

Competence and skills
- write small functional programs for various applications
- structure programs in a way that makes them easy to understand and modify, by appropriate application of data types, abstraction, and code reuse
- implement effective tests for functional programs with help of suitable tools

Judgement and approach
- show the ability, in various contexts, to judge which programming techniques are most appropriate for solving the problem at hand

Course content

This is an introductory course in programming, and uses a functional language. The primary goal is to enable students to write small programs, while introducing some of the fundamental concepts of computer science. Secondary goals are to provide orientation regarding the courses to come (particularly data structures, algorithms, digital circuits, formal methods and programming language courses).

Concrete topics encountered in the course include:

- values, types and functions
- compound data types (lists, tuples, and user-defined types)
- higher-order functions
- using abstraction to avoid repetitive programming
- recursion and recursive data types
- efficient and inefficient programs
- input-output
- verification with the help of testing

Sub-courses

1. **Written examination** *(Skriftlig tentamen),* 4.5 credits
   Grading scale: Pass with distinction (5), Pass with credit (4), Pass (3) and Fail (U)

2. **Laboratory work** *(Laboration),* 3 credits
   Grading scale: Pass (G) and Fail (U)

Form of teaching

The teaching is based on lectures, group meetings, and supervised lab sessions.

Language of instruction: Swedish and English
The course might be given in Swedish or English.

Assessment

The course is examined by an individual written exam carried out in an examination hall at the end of course, and a number of compulsory assignments typically carried out in groups of 2-3 students.

If a student, who has failed the same examined component twice, wishes to change examiner before the next examination, a written application shall be sent to the department responsible for the course and shall be granted unless there are special reasons to the contrary (Chapter 6, Section 22 of Higher Education Ordinance).

In cases where a course has been discontinued or has undergone major changes, the student shall normally be guaranteed at least three examination occasions (including the ordinary examination) during a period of at least one year from the last time the course was given.

Grades

The grading scale comprises: Pass with distinction (5), Pass with credit (4), Pass (3) and Fail (U).
To pass the course, students must receive a passing grade in both modules. The grade for the entire course will be determined by the written exam.

Course evaluation

The course is evaluated through meetings both during and after the course between teachers and student representatives. Further, an anonymous questionnaire is used to ensure written information. The outcome of the evaluations serves to improve the course by indicating which parts could be added, improved, changed or removed.

Additional information

The course is a joint course together with Chalmers.

Course literature to be announced the latest 8 weeks prior to the start of the course.

Students with previous programming knowledge are encouraged to take the course DIT143 Functional Programming instead.

The course replaces the course DIT440, 7.5 credits. The course cannot be included in a degree which contains DIT143 or DIT440. Neither can the course be included in a degree which is based on another degree in which the course DIT143 or DIT440 is included