



# Syllabus

for course at first level

**Programming Paradigms**  
**Programmeringsparadigm**

**7.5 Higher Education**  
**Credits**  
**7.5 ECTS credits**

<b>Course code:</b>	DA3012
<b>Valid from:</b>	Spring 2013
<b>Date of approval:</b>	2012-10-08
<b>Department</b>	Department of Mathematics (incl. Math. Statistics)
<b>Main field:</b>	Computer Science
<b>Specialisation:</b>	G1F - First cycle, has less than 60 credits in first-cycle course/s as entry requirements

## Decision

This syllabus has been approved by the Board of the Faculty of Science at Stockholm University, September 27, 2006, and revised May 19, 2008.

## Prerequisites and special admittance requirements

### Course structure

Examination code	Name	Higher Education Credits
THEO	Theory	3
LABO	Practical Exercises	4.5

### Course content

a. The course covers basic programming language knowledge and the most important programming paradigms through a systematic review.

Programming language knowledge represents

- programming paradigms and programming language history
- language definition: syntax, semantics
- language design principles: generality, orthogonality, uniformity
- language translation: interpretation, compilation, linking
- programming principles: modularity, programming style
- programming concepts

The programming paradigms part brings up:

- functional programming: declarative programming with mathematical functions as in Haskell, names denoting values, higher-order functions, type classes, strict and non-strict evaluation, streams, input pattern matching
- logic programming: declarative programming with logical predicates as in Prolog, names denoting values, unification, resolution, non-deterministic programming
- imperative programming: programming with commands that can change values in memory locations as in C
- object oriented programming: imperative programming with objects and classes as in Java, methods, instantiation, inheritance, polymorphism

- Internetprogramming: the web as an example of client-server computing, differences between CGI, RMI, applets and servlets.

b. The course consists of the following items:

- Theory, 4.5 HECs
- Practical Exercises, 3 HECs

### **Learning outcomes**

It is expected that the student after taking the course will be able to:

- judge which programming paradigm and programming language that is suitable for fulfilling a certain task
- follow good style in the chosen programming paradigm
- solve programming problems in different programming paradigms

### **Education**

The education consists of lectures and practical exercises.

Participation in practical exercises is compulsory. An examiner may rule that a student is not obliged to participate in certain compulsory education, if there are special grounds for this, after consultation with the relevant teacher.

### **Forms of examination**

a. Examination for the course is in the following manner: measurement of knowledge of the element Theory takes place through written and/or oral examination, and of the element Practical Exercises through written and/or oral presentations.

b. Grading is carried out according to a 7-point scale related to learning objectives:

- A = Excellent
- B = Very Good
- C = Good
- D = Satisfactory
- E = Sufficient
- Fx = Fail
- F = Fail

c. Grading criteria for the course will be distributed at the start of the course.

d. A minimum grade of E is required to pass the course, together with completion of all compulsory education.

e. Students who fail to achieve a pass grade in an ordinary examination have the right to take at least further four examinations, as long as the course is given. The term “examination” here is used to denote also other compulsory elements of the course. Students who have achieved a pass grade on an examination may not retake this examination in order to attempt to achieve a higher grade. Students who have failed to reach a pass grade on two occasions have the right to request that a different teacher be appointed to set the grade of the course. A request for such appointment must be sent to the departmental board.

### **Interim**

Students may request that the examination is carried out in accordance with this syllabus even after it has ceased to apply. This right is limited, however, to a maximum of three occasions during a two-year-period after the end of giving the course. A request for such examination must be sent to the departmental board.

### **Misc**

The course is a component of the Master’s Programme in Computer Science, and it can also be taken as an individual course.

### **Required reading**

Course literature is decided by the departmental board and is described in an appendix to the syllabus.