

System Programming (C003776)

Course size (nominal values; actual values may depend on programme)

Credits 6.0 Study time 180 h Contact hrs 60.0 h

Course offerings and teaching methods in academic year 2020-2021

A (semester 1)	Dutch	lecture	30.0 h
		project	5.0 h
		group work	5.0 h
		seminar: practical PC room classes	20.0 h

Lecturers in academic year 2020-2021

De Turck, Filip	TW05	lecturer-in-charge
Volckaert, Bruno	TW05	co-lecturer

Offered in the following programmes in 2020-2021

	crdts	offering
Bachelor of Science in Computer Science	6	A
Bachelor of Science in Mathematics	6	A
Master of Science in Bioinformatics (main subject Systems Biology)	6	A

Teaching languages

Dutch

Keywords

Procedural programming, C programming language, hybrid programming language, C++ , software design paradigms

Position of the course

After prior introductory courses in programming, the objective of this course is to widen the knowledge and understanding of programming languages, as well as to treat the basic principles of paradigms for software design in a more generic way : procedural, object oriented and aspect oriented.

The major objective is to give an extended overview of current paradigms for software engineering. It fits in the bachelor curriculum since programming techniques are put in a broader context. Students are prepared for later courses which are more about development methodology and also for carrying out realistic development projects in the final phase of the bachelor curriculum.

Contents

- 1 Procedural programming (using C) and hybrid languages (using C++).
This implies following skills:
 - using pointers
 - reference types versus value semantics
 - operator overloading
 - inheritance, both single and multiple
 - separation of interface and implementation, illustrating a weakness of C++
 - generic programming and data abstraction
 - the STL library
 - exception handling
- 2 Paradigms of software design
 - object oriented paradigm
 - a conceptual discussion of object orientation
 - the limitations of the paradigm
 - platforms for support of large software projects with multiple programmers
 - Version Systems

- Generation of Makefiles
 - Usage of Ant and Maven
 - overview of currently important software technologies
 - Domotics, Telecommunications, Backend platforms, Proces Control
 - Application on portable devices (PDA,s Smart phones, etc)
- 3 3. A software design project (groups of 2 students) focusing on software development in C/C++

Initial competences

A good knowledge of at least one programming language, preferably an object oriented language (Java)
 Initial experience with the basic principles of object orientation
 Some knowledge of computer architecture
 Experience with using a computer
 These objective are met by the courses "Programming" and "Object Oriented Programming" which come earlier in the curriculum

Final competences

- 1 A good view on several paradigms for software design.
- 2 A good knowledge of several programming languages.
- 3 Insight in the available platforms for support of large projects.
- 4 An overview of currently important software technologies.

Conditions for credit contract

Access to this course unit via a credit contract is determined after successful competences assessment

Conditions for exam contract

This course unit cannot be taken via an exam contract

Teaching methods

Group work, lecture, project, seminar: practical PC room classes

Learning materials and price

The presentations used in the plenary sessions can be downloaded from the e-learning environment Ufora.
 Problems and training material for the lab sessions (via Ufora), with individual feedback.Extra materials via Ufora.

Recommended reference book: "A Book on C", Kelley en Pohl, 4e editie, Addison-Wesley
 will be advised. Distribution of the book: to be organised by the students' association

Price of the material :
 The material that can be downloaded from Ufora: free
 Reference book: 40 Euro

References

De Programmeertaal C, vierde editie (Al Kelley, Ira Pohl), ISBN: 90.430-0497.9
 De programmeertaal C++ (Bjarne Stroustrup), ISBN: 90.430-0231.3

Course content-related study coaching

An e-learning environment (including discussion fora supporting the building of a community)
 Lab sessions : assistants are available to help the students during these sessions and give feedback on the submitted solutions
 Teacher and assistants can be reached by e-mail

Evaluation methods

end-of-term evaluation and continuous assessment

Examination methods in case of periodic evaluation during the first examination period

Open book examination, skills test

Examination methods in case of periodic evaluation during the second examination period

Open book examination, skills test

Examination methods in case of permanent evaluation

Assignment, skills test

Possibilities of retake in case of permanent evaluation

examination during the second examination period is possible in modified form

Extra information on the examination methods

Periodical evaluation: exam on PC (study material can be used).

Non-periodical evaluation: scores of the software project and reports (groups of 4 students)

Calculation of the examination mark

First examination chance: periodical evaluation (75%) + non-periodical evaluation (25%).

Second examination chance: periodical evaluation (100%).