linfo1104

2019

In view of the health context linked to the spread of the coronavirus, the methods of organisation and evaluation of the learning units could be adapted in different situations; these possible new methods have been - or will be - communicated by the teachers to the students.

5 credits	30.0 h + 30.0 h	Q2

Teacher(s)	Van Roy Peter ;
Language :	French
Place of the course	Louvain-la-Neuve
Prerequisites	This course assumes that the student already masters basic programming skills targeted by courses LINFO1101 or LEPL1401 and concepts on algorithmics and simple data structures covered by course LEPL1402. *The prerequisite(s) for this Teaching Unit (Unité d'enseignement – UE) for the programmes/courses that offer this Teaching Unit are specified at the end of this sheet.*
Main themes	• Programming paradigms (including functional, object-oriented, dataflow, and multi-agent) • Formal semantics and reasoning techniques • Procedural abstraction (higher-order programming) • Data abstraction (including object orientation and abstract data types) • Concurrent programming (including shared state, message passing, and deterministic dataflow) • Symbolic programming (including algorithm design)
Aims	1 Regarding the learning outcomes of the program of Bachelor in Engineering, this course contributes to the development and the acquisition of the following learning outcomes: • LO 1.1, 1.2 • LO 2.3, 2.4, 2.5, 2.6, 2.7 • LO 4.2, 4.3, 4.4 Given the learning outcomes of the "Bachelor in Computer Science" program, this course contributes to the development, acquisition and evaluation of the following learning outcomes: • S1.I2., S1.I3, S1.I5 • S2.2., S2.3, S2.4 • S5.3, S5.4., S5.5. Students completing successfully this course will be able to • specify the problems, divide them into their basic steps, and design algorithms and abstractions to solve them; • choose the right programming paradigm and write a program in this paradigm to solve a problem; • use formal semantics to explain the accuracy of the program; • write small concurrent programs in the deterministic dataflow paradigm. • think using abstractions (reason correctly on a system that includes several layers of abstractions, and define new abstractions to simplify the resolution of a problem) - - - - *The contribution of this Teaching Unit to the development and command of the skills and learning outcomes of the programme(s) can be accessed at the end of this sheet, in the section entitled "Programmes/courses offering this Teaching Unit".*
Evaluation methods	**Due to the COVID-19 crisis, the information in this section is particularly likely to change.** • Dispensatory test 25% (around week 7) • Project 25% • Final exam 50% (or 75% if redoing test part) The project is mandatory and is done during the quadrimester. It can only be done only once and it counts for the whole academic year.
Teaching methods	**Due to the COVID-19 crisis, the information in this section is particularly likely to change.** • Weekly lectures • Practical lab sessions in the computer room every week, to solve simplified problems using concepts explained during the lectures. • One major design and programming project to apply these concepts to a more complex application

Content	The goal of this course is to widen and deepen the programming knowledge acquired in preceding courses. This course will treat the following subjects: • The course will give a uniform framework for all programming concepts, organised as programming paradigms. • The course will give a formal semantics and reasoning techniques for all paradigms. • Higher-order programming will be used as organising principle for the construction of procedural abstractions. • Concurrent programming will be presented in its three principal forms, namely shared-state concurrency, message-passing concurrency, and deterministic dataflow concurrency. • Data abstraction will be presented in its general form and with its two principal derived forms, namely object-oriented programming and abstract data types. • Symbolic programming and algorithm design. • Several programming paradigms will be presented in the course, including functional programming, object-oriented programming, deterministic dataflow programming, and multi-agent programming. Examples of practial applications will be given for all concepts and all paradigms.
Inline resources	LINFO1104 Moodle.
Bibliography	Peter Van Roy et Seif Haridi, PROGRAMMATION: Concepts, techniques et modèles, Dunod, 2007 Peter Van Roy et Seif Haridi, Concepts, Techniques, and Models of Computer Programming, MIT Press, 2004
Other infos	Prequisite: • Second-year programming course Informatique 2 (LEPL1402) or equivalent.
Faculty or entity in charge	INFO

Programmes containing this learning unit (UE)

Program title	Acronym	Credits	Prerequisite	Aims
Specialization track in Computer Science	LINFO100P	5		🔍
Bachelor in Computer Science	SINF1BA	5	LINFO1101	🔍