LFSAB1402

2016-2017

Informatics 2

5.0 credits	30.0 h + 30.0 h	1q

Teacher(s) :	Van Roy Peter ;
Language :	Français
Place of the course	Louvain-la-Neuve
Inline resources:	> http://icampus.uclouvain.be/claroline/course/index.php?cid=FSAB1402
Prerequisites :	LFSAB1401 or LSINF1101 or equivalent course *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 :	The principal themes covered by this course are: -- Techniques for deriving the computational complexity of an algorithm -- Techniques for reasoning about programs -- Object-oriented modeling -- Linear and tree-like data structures -- Recursive algorithms -- Implementation in high level programming language of medium-sized programs -- Methods for testing and validating programs
Aims :	Contribution of the course to the program objectives 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. Specific learning outcomes of the course More precisely, at the end of the course the students will be able to -- make a choice between several data representations and algorithms to process them, -- reason on program fragments: algorithmic complexity and efficiency of the programs that implement them, reasoning with recursion, -- apply the principles of object-oriented modeling, -- design and apply methods for program testing. Students will have developed skills and operational methodology. In particular, they have developed their ability to: -- analyze a problem of medium-sized to provide an IT solution and implement it in a high level programming language.

	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 :	The evaluation has 2 components: an intermediary evaluation during the semester and a final exam at the end of the semester (written exam). The final mark is a combination of the scores in these two evaluations
Teaching methods :	The chosen teaching method relies on active student participation in their own learning process. The specific modalities of the active learning approach used in the course are left to the initiative of the course teachers, within the framework of the pedagogical choices made by EPL.
Content :	Data abstraction Linear data abstractions (stacks, queues, lists, etc.) and their applications Techniques for representing linear data abstractions Object-oriented modeling (inheritance, composition, and reuse) Preconditions, postconditions, invariants Reasoning techniques (deduction rules, termination proofs, ...) Basics of computational complexity Derivation of the temporal complexity of an algorithm Derivation of the spatial complexity of a data structure Recursive formulation of a solution and recursive algorithms Tree-like data abstractions (binary trees) and their applications Techniques for representing tree-like data abstractions Quantified measurements of program efficiency Design and implementation of methods for testing and validating programs
Bibliography :	Workfiles for each of the parts (available on the website and in printed version) 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
Faculty or entity in charge:	BTCI

Programmes / formations proposant cette unité d'enseignement (UE)

Intitulé du programme	Sigle	Credits	Prerequis	Acquis d'apprentissage
Bachelor in Computer Science	SINF1BA	5	LSINF1101	🔍
Bachelor in Engineering	FSA1BA	5	LFSAB1401	🔍
Minor in Computer Sciences	LINFO100I	5	LSINF1101	🔍