LSINF1101

2013-2014

Introduction to programming

6.0 credits	30.0 h + 30.0 h	1q

Teacher(s) :	Pecheur Charles ; Bonaventure Olivier ;
Language :	Français
Place of the course	Louvain-la-Neuve
Inline resources:	> http://icampus.uclouvain.be/claroline/course/index.php?cid=lsinf1101
Main themes :	-- Systematic construction of algorithms starting from a set of primitive objects and a set of primitive actions upon these objects. -- Systematic translation of an algorithm into a programming language (Java). -- Representations of numbers and characters -- Syntax and data types in a programming language
Aims :	Students completing successfully this course will be able to -- construct complex algorithms by decomposing the problem into successive subproblems to arrive at basic objects and primitive actions -- describe representation of numeric and character data and how precision, round-off and data conversion can affect numeric calculations -- explain the basic concepts describing the operation of a central unit and a machine language and indicate their role in executing a simple instruction -- exploit the basic elements of a high-level language (literals, tables, methods, ...) to implement simple algorithms -- explain the gradation between the binary representation of data in computers, the machine language and programming in a high-level language. Students will have developed skills and operational methodology. In particular, they have developed their ability to -- have a critical look at their achievements and justify the steps of the reasoning that led them there *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 :	A middle evaluation takes place in the middle of the first period. The note for this exam takes account of 1/3 of the final grade only if it is greater than the examination mark. The end of first period exam aims to test not only the knowledge, but also the ability to apply this knowledge to write simple Java programs, but correct
Teaching methods :	The methods used will encourage active student learning. The detailed arrangements for implementing an active student participation in learning are left to tenured, in accordance with the educational guidelines of the Faculty.
Content :	The first chapter of the course addresses the construction of algorithms. Primitive objects are given with primitive operations upon them. It is shown how arbitrarily complex algorithms acting on the objects can be built using sequential, conditional, and iterative composition of algorithms. The emphasis is put on specifications and on the use of assertions to derive correct code. The Java programming language is used in a controlled way as a tool to make the algorithms amenable to execution by a computer. The second chapter is concerned with the representation of data. Decimal, binary, octal and hexadecimal notations are reviewed as well as conversion algorithms from one notation to another. The two's complement representation for negative numbers is explained as well as the representation of floating point numbers. ASCII and Unicode representations for characters are described and discussed. In the third chapter, the CPU of a simple computer is described together with its machine language. Students are taught how to write programs in this simple machine language. The design of subprograms with standard conventions for subprogram calls and returns as well as parameter passing is discussed in details as it allows the student to better understand procedure calls and parameter passing methods in higher level programming languages. The fourth chapter addresses programming languages concepts in a more systematic way. The Java programming language is used to illustrate the concepts but no attempt is made to provide a global overview of Java. In contrast, a limited number of topics are studied with care and in details : primitive data types, literals, variables, arrays, the String class, static methods, expressions,

a small subset of composed statements, primitive type conversions, method overloading. Classical algorithms for searching and sorting are build with this subset of Java.

Bibliography :	The course is based on the following reference book (in English): D. Eck, Introduction to Programming Using Java, Sixth Edition, 2013, http://math.hws.edu/javanotes/
Cycle and year of study :	> Master [120] in Information and Communication Science and Technology > Master [120] in Linguistics > Bachelor in Economics and Management > Bachelor in Mathematics > Bachelor in Computer Science
Faculty or entity in charge:	INFO