

Due to the COVID-19 crisis, the information below is subject to change, in particular that concerning the teaching mode (presential, distance or in a comodal or hybrid format).

5 credits

30.0 h + 30.0 h





Q2

Teacher(s)	Dupont Pierre ;
Language :	French
Place of the course	Louvain-la-Neuve
Main themes	<ul style="list-style-type: none"> <li>• Design and implementation of iterative or recursive algorithms: path, counting, sorting, searching in collections</li> <li>• Computational complexity</li> <li>• Basic data structures: arrays, stacks, queues, linked lists</li> <li>• Recursive data structures: tree structures, binary search trees</li> <li>• Invariants</li> </ul>
Aims	<p>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:</p> <ul style="list-style-type: none"> <li>• S1.I2, S1.I3</li> <li>• S2.2-4</li> <li>• S6.2</li> </ul> <p>Students who have successfully completed this course will be able to:</p> <ul style="list-style-type: none"> <li>• justify a choice between several algorithmic solutions to solve a given problem,</li> <li>• analyze algorithms, iterative or recursive, to represent and manipulate collections and to propose variants thereof,</li> <li>1 • choose, design and use data structures, including recursive,</li> <li>• give a reasoned estimate of the time complexity of iterative algorithms and the spatial complexity of data structures;</li> <li>• reasoning about properties of algorithms or data structures in terms of invariants.</li> </ul> <p>Students will have developed methodological and operational skills. In particular, they have developed their ability to:</p> <ul style="list-style-type: none"> <li>• to take a critical look and make a reasoned analysis of a solution or set of solutions that could be made to a given problem by setting quality criteria,</li> <li>• realize small programs using conventional algorithms and data structures.</li> </ul> <p>-----</p> <p><i>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".</i></p>
Evaluation methods	<p><b>Due to the COVID-19 crisis, the information in this section is particularly likely to change.</b></p> <p>A note of PARTICIPATION reflects the involvement of the student during the year for solving the practice problems on Ingenious and when implementing the projects.</p> <p>In the first session, the participation grade is worth 20% of the final grade + 80% for the final exam (closed book). The participation mark can not be reassessed.</p> <p>In the second session, participation grade and the final exam are worth respectively 10 % and 90% of the overall score.</p> <p>The final exam is, by default, a written exam (on a computer or, when appropriate, on paper). These evaluation rules are subject to possible updates due to the sanitary situation. In particular, the relative weights between the participation grade (or specifically the projects) and the final exam could be adapted. Such possible updates would be notified to the students by a general announcement posted on the Moodle site of this course.</p>

Teaching methods	<p>Due to the <b>COVID-19</b> crisis, the information in this section is particularly likely to change.</p> <ul style="list-style-type: none"> <li>• Lectures</li> <li>• Practical sessions on the Inginious server</li> <li>• 2 mini-projects at the end of the semester</li> </ul> <p>By default, <i>lectures</i> can be followed <b>face to face</b> in the auditorium announced in the official schedule. Depending on the number of registered students and the evolution of the sanitary situation, students will be able to follow the lectures as well <b>remotely</b> on Teams.</p> <p>Practice problems and projects are submitted <b>on line</b> and evaluated on the Inginious platform.</p> <p><b>Practical sessions</b> to help students to solve the practical problems are given, by default, <b>face to face</b>. Depending on the sanitary situation, these sessions could be given, <b>partially or fully, on line</b> on Teams.</p>
Content	<p>Algorithmics is concerned with solving problems by implementing sequences of elementary operations according to a predefined process or procedure leading to a solution.</p> <p>This discipline is both abstract and put into practice through programs (e.g. implemented in Python) and run on a computer.</p> <ul style="list-style-type: none"> <li>• Time and space complexity</li> <li>• Search algorithms through arrays</li> <li>• Abstract data types, data structures: stacks, queues, dynamic arrays, linked lists</li> <li>• Sorting algorithms</li> <li>• Recursion</li> <li>• Recursive data types</li> <li>• Computational complexity of recursive algorithms, recurrence equations</li> <li>• Binary trees and dictionnaires</li> <li>• Invariants</li> </ul>
Inline resources	<a href="http://moodleucl.uclouvain.be/course/view.php?id=9010">http://moodleucl.uclouvain.be/course/view.php?id=9010</a>
Bibliography	Il n'y a pas d'ouvrage de référence obligatoire mais, à titre complémentaire, des ouvrages sont recommandés sur le site Moodle.
Faculty or entity in charge	INFO

### Force majeure

Teaching methods	Lectures are given <b>online</b> and can be followed <b>remotely</b> . Practical exercices are submitted <b>online</b> on the Inginious platform.
Evaluation methods	The final exam is <b>online</b> on the Inginious platform.

Programmes containing this learning unit (UE)				
Program title	Acronym	Credits	Prerequisite	Aims
Master [120] in Linguistics	LING2M	5		
Approfondissement en sciences et technologies de l'information et de la communication (pour seule réinscription)	APPSTIC	6		
Minor in Computer Sciences	MINSINF	5		
Bachelor in Computer Science	SINF1BA	5		
Minor in numerical technologies and society	MINSTIC	5		