





5.00 credits	30.0 h + 15.0 h	Q1
--------------	-----------------	----

Teacher(s)	Schaus Pierre ;
Language :	English > French-friendly
Place of the course	Louvain-la-Neuve
Main themes	<ul style="list-style-type: none"> <li>• tree research exploration</li> <li>• branch and bound</li> <li>• relaxation (Lagrangian) and calculation of terminals</li> <li>• local search</li> <li>• mathematical programming</li> <li>• constraint programming</li> <li>• graph algorithms</li> <li>• wide neighborhood research</li> <li>• dynamic programming</li> <li>• greedy algorithms and approximation algorithms</li> <li>• multi-criteria optimization</li> <li>• optimization without derivative</li> <li>• comparisons of algorithms</li> </ul> <p>These methods will be applied to real problems like vehicle routing, scheduling and rostering confection, network design, scheduling and scheduling, etc..</p>
Learning outcomes	<p><b>At the end of this learning unit, the student is able to :</b></p> <p>Given the learning outcomes of the "Master in Computer Science and Engineering" program, this course contributes to the development, acquisition and evaluation of the following learning outcomes:</p> <ul style="list-style-type: none"> <li>• INFO1.1-3</li> <li>• INFO2.3-5</li> <li>• INFO5.3-5</li> <li>• INFO6.1, INFO6.4</li> </ul> <p>Given the learning outcomes of the "Master [120] in Computer Science" program, this course contributes to the development, acquisition and evaluation of the following learning outcomes:</p> <p>1</p> <ul style="list-style-type: none"> <li>• SIN1.M4</li> <li>• SIN2.3-5</li> <li>• SIN5.3-5</li> <li>• SIN6.1, SIN6.4</li> </ul> <p>Students completing this course successfully will be able to</p> <ul style="list-style-type: none"> <li>• explain the algorithms for solving discrete optimization problems by describing precisely specifying the problems they solve, indicating their advantages, disadvantages and limitations (computing time, accuracy, problems of scaling , etc.),</li> <li>• identify the algorithms that apply to a discrete optimization problem they are facing and make an arguedchoice among them ,</li> <li>• implement algorithms for solving discrete optimization problems.</li> </ul>

Evaluation methods	<p><b>January:</b></p> <p>For the first session, the global grade for the course is solely based on the grades of the computing projects, submitted and evaluated during the semester.</p> <p><b>August:</b></p> <p>For the second session, previously submitted projects will not be re-evaluated and cannot be resubmitted. Instead, students will be assigned a new individual programming project after the June session. This project will require a written report, and, if deemed necessary by the instructor, an interview about the project may also be conducted also to verify that all theoretical concepts are well understood.</p> <p><b>Use of Generative AI in Course Assignments (# Source: Written with thehelp C hatGPT and Prof. Schaus &amp; Dupont for prompting)</b></p> <p>Projects are individual.</p> <p>Nevertheless, In this course, we recognize the evolving nature of technology and the potential benefits of using generative AI tools in the programming process. However, academic honesty and originality remain paramount. To that end:</p> <ul style="list-style-type: none"> <li>• <b>Generative AI Usage:</b> Students are permitted to use generative AI tools to assist with their assignments. Such tools can provide inspiration, suggest coding approaches, or help troubleshoot issues.</li> <li>• <b>Original Work:</b> While AI can be a tool, it should not be the sole author of your assignment. Your submission should be primarily your own work. Directly copying and pasting solutions from AI outputs without understanding or modification is discouraged. Similarly, collaborating with fellow students is a valuable part of the learning process, but directly copying another student's work is considered plagiarism.</li> <li>• <b>Source Indication:</b> Whenever you use generative AI to assist in your assignment, you are required to indicate so by providing a brief comment in your code on how the AI was used. For example:</li> </ul> <pre>python  # Used AI to suggest optimization for this loop.  for i in range(10): ...</pre> <p>Failure to adhere to these guidelines may result in a reduction of marks or other academic penalties. The same consequences will hold for a student that voluntarily shares his code or make available to other students (this includes sharing your code on a public or private repository). If deemed necessary by the instructor, an interview about the projects may also be conducted.</p>
Teaching methods	<p>The presentation of the algorithms will be either proposed in the form of lectures, videos or reading and will be accompanied by practical work (assignments / micro-projects) requesting the implementation algorithms to solve a practical optimization problem and the writing of reports.</p>
Content	<p>This course introduces a range of techniques for exact and approximate solutions to combinatorial optimization problems. Among the techniques:</p> <ul style="list-style-type: none"> <li>• dynamic programming</li> <li>• branch and bound</li> <li>• linear programming</li> <li>• Lagrangian relaxation</li> <li>• column generation</li> <li>• local search</li> <li>• constraint programming</li> <li>• graph algorithms: flows</li> <li>• comparisons of optimization algorithms</li> </ul> <p>These methods will be applied to real problems like vehicle routing, scheduling and rostering confection, network design, scheduling and scheduling, etc..</p>
Inline resources	<p><a href="https://moodle.uclouvain.be/course/view.php?id=1474">https://moodle.uclouvain.be/course/view.php?id=1474</a></p>
Other infos	<p>Background: a good knowledge of data structures and algorithms (for instance obtained by having followed the course LINFO121) and a good knowledge of Java language</p>

Faculty or entity in charge	INFO
-----------------------------	------

<b>Programmes containing this learning unit (UE)</b>				
Program title	Acronym	Credits	Prerequisite	Learning outcomes
Master [120] in Data Science : Statistic	DATS2M	5		
Master [120] in Computer Science and Engineering	INFO2M	5		
Master [120] in Computer Science	SINF2M	5		
Master [120] in Data Science Engineering	DATE2M	5		
Master [120] in Data Science: Information Technology	DAT12M	5		