UCLouvain

linfo2365

2022

Constraint programming

5.00 credits	30.0 h + 15.0 h	Q2

Teacher(s)	Schaus Pierre ;
Language :	English > French-friendly
Place of the course	Louvain-la-Neuve
Main themes	• Constraints and domains • Practical aspects of c onstraint solvers • Constraint Satisfaction Problems (CSP) • Models and languages for constraint programming • Methods and techniques for constraint solving (consistency, relaxation, optimization, search, linear programming, global constraints, ...) • Search techniques and strategies • Problem modelling and resolution • Applications to differents problem classes (e.g. planification, scheduling, ressource allocation, economics, robotics)
Learning outcomes	**At the end of this learning unit, the student is able to :** 1 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: • INFO1.1-3 • INFO2.2-4 • INFO5.4-5 • INFO6.1, INFO6.4 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: • SINF1.M4 • SINF2.2-4 • SINF5.4-5 • SINF6.1, SINF6.4 Students completing successfully this course will be able to • explain and apply techniques for solving Constraint Satisfaction Problems • solve simple problems involving CSP • explain foundations of models and languages for constraint solving • identify problem classes where constraint programming can be apply successfully • model simple problems in the form of constraints, and express these models in a constraint programming language, including search strategies. Students will have developed skills and operational methodology. In particular, they have developed their ability to: • master rapidly a new programming language; • use technical documents to deepen their knowledge of a topic.
Evaluation methods	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. The projects are not evaluated again for the second session and may not be resubmitted. The grades for projects are kept as such representing 50% and the other 50% are evaluated with a written exam, or when appropriate, on a computer. Projects are invididual. It means that any source code of a project estimated to be - copied or inspired by the one of another student, or - copied or inspired by a source code found on the internet or another source, will result in a zero grade for the student at the projects and the exam The same consequences will hold for a student that voluntarily shares his code or make available to other students.

Teaching methods	Students will follow a MOOC on the EdX plateform (videos) and there will be programming exercises and quizzes graded on inginious.
Content	• Constraint Programming : a Declarative Programming paradigm • Architecture of a constraint programming solver • Global contraints and implementation techniques (incrementality, etc) • Search techniques and strategies • Combinatorial optimization problem modeling and solving • Applications to different problem classes (e.g. planification, scheduling, resource allocation, economics, robotics)
Inline resources	https://moodleucl.uclouvain.be/course/view.php?id=9158 www.minicp.org
Bibliography	Le site www.minicp.org + lectures suggérées pendant le semestre
Other infos	A good background in data-structure and algorithms is required to follow this course and a good knowledge of Java language
Faculty or entity in charge	INFO

Programmes containing this learning unit (UE)				
Program title	Acronym	Credits	Prerequisite	Learning outcomes
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	DATI2M	5		