

FICHE ECTS

Composante (dépt ou centre)	DGEI
Code ECTS	I5IRSD12
Crédits ECTS	4
Enseignant responsable de l'UF	P. Esquirol
Section CNU	27-61
Libellé de l'UF (français)	Analyse de données prescriptive
Libellé de l'UF (anglais)	Prescriptive Analytics
Semestre :	Semestre 1 <input checked="" type="checkbox"/> Semestre 2 <input type="checkbox"/> Annuel <input type="checkbox"/>

DESCRIPTION GENERALE

VERSION FRANÇAISE

Objectifs, finalités ¹ (max 1000 caract.)

Ce cours adresse des modèles de traitement efficace des données rencontrées dans des problèmes industriels à caractère combinatoire. Les modèles sont basés sur l'inférence logique et l'optimisation : les problèmes de satisfaction de contraintes (CSP), les modèles à base de clauses disjonctives booléennes (SAT) et la programmation linéaire en nombres entiers (PLNE).

Pour la partie CSP, les étudiants doivent connaître les principales techniques de propagation et stratégies de résolution et se familiariser à travers les travaux pratiques avec des outils de programmation intégrant des algorithmes généraux de résolution (ex : CPLEX).

Dans la partie modélisation SAT, les étudiants appliquent différentes techniques et heuristiques de propagation de contraintes sur des modèles SAT (DPPLL, Graphes d'Implication, Analyse de conflits, Two-watched literals algorithm). Différents problèmes combinatoires classiques (coloration, affectation de ressources, ordonnancement) servent de cas pratiques pour s'entraîner sur l'encodage SAT.

Pour la partie PLNE, les étudiants doivent modéliser des problèmes industriels sous forme de programme linéaire en nombre entiers, et les résoudre via des algorithmes de branchement ou des méthodes de décomposition en utilisant des outils de programmation (CPLEX).

On conseille d'éviter des verbes vagues pour lesquels il sera difficile de vérifier si l'objectif est atteint (connaître, comprendre, apprendre, savoir...). On peut s'inspirer des verbes de la taxonomie de Bloom.

VERSION ANGLAISE

Objectives (max 1000 charact.)

This course addresses several efficient models for processing data encountered in industrial combinatorial problems. These models are based on logical inference and mathematical optimisation techniques : constraint satisfaction problems (CSPs), boolean satisfiability (SAT) and integer linear programming (ILP).

In the first part (CSPs), students are expected to understand and to be able to apply the main constraint propagation techniques and solving strategies, by hand, but are also initiated to programming tools that integrate general solvers (ex : CPLEX) during practical works.

In the SAT modeling part of this course, students are initiated to some propagation and heuristic solving techniques used in SAT solvers (DPPLL, Implication Graphs, Conflict Analysis, Two-watched literals algorithm). Various applications problems such as allocation, graph colouring, scheduling serve as training examples for SAT encoding.

In the last part (MILPs), students will have to translate industrial problems into mixed-integer linear programs, then to solve them efficiently using branching algorithms or decomposition methods, embedded in existing tools such as CPLEX.

Contenu (max 1000 caract.)	Description (max 1000 caract.)
<p>Problèmes de satisfaction de contraintes : satisfaisabilité et consistance. Formulation des contraintes en intension et en extension. Réseaux de contraintes : algorithmes de nœud, arc et chemin-consistance. Arc-consistance généralisée. Bornes-consistance. Contraintes globales (Alldiff, Sum, Cardinality, Disjunctive, Cumulative).</p> <p>Dans la partie SAT, <Mohamed> ...</p> <p>Programmation Linéaire en Nombres Entiers : modélisation et résolution. Modélisation de problèmes industriels en PLNE. Algorithmes de résolution standards : branch & bound, branch & Cut. Méthodes de décomposition : génération de colonnes, décomposition de Benders.</p>	<p>Constraint Satisfaction Problems: satisfiability and consistency. Implicit and explicit constraint formulations. Constraint Networks : node, arc and path consistency algorithms. Generalised Arc-Consistency. Bound-consistency. Global constraints (Alldiff, Sum, Cardinality, Disjunctive, Cumulative).</p> <p><Mohamed> ...</p> <p>Integer Linear Programming: design and resolution. Industrial problems modelling in MILP. Standard solving algorithms: branch & bound, branch & cut. Decomposition methods: column generation, Benders' decomposition.</p>
Vous pouvez aussi mentionner ci-dessus les documents remis aux étudiants	

Recommandation (max 1000 caract.)	Recommendation (max 1000 caract.)
Principales difficultés habituellement rencontrées par les étudiants	

Pré-requis (Code UF + intitulé, sinon notions nécessaires) (max 200 caract.)	Necessary knowledge (UF Code + title, or required knowledge) (max 200 caract.)
Algorithmics & programming (I2MIIF11, I2MIIF21). Fundamentals in Computer Science (I4IRIF11), Intelligent Systems (I4IRSD11).	Algorithmics & programming (I2MIIF11, I2MIIF21). Fundamentals in Computer Science (I4IRIF11), Intelligent Systems (I4IRSD11).

Organisation, méthodes pédagogiques			Organisation, teaching methods		
Horaires présentiel (tel que l'enseignement est comptabilisé)		Travail personnel²	Contact hours		Personal work
CM			Lectures		
TD	22,5		Tutorials	22,5	
TP	27,5		Lab work	27,5	
Projet			Project		
Examen formatif			Coursework		
Examen certificatif			Exam		

Format d'enseignement :	Présentiel <input checked="" type="checkbox"/>	Distanciel <input type="checkbox"/>	Hybride <input type="checkbox"/>
--------------------------------	--	-------------------------------------	----------------------------------

DESCRIPTION COMPLEMENTAIRE

Modalités d'évaluation (max 1000 caract.)		Assessment (max 1000 charact.)	
<i>Comment évaluez-vous que ces objectifs sont atteints ?</i>			
L'examen écrit est fondé sur des exercices qui testent que les différents concepts présentés en cours sont compris.		The written exam is based on exercises that test that the different concepts presented in class are understood.	
Examen écrit <input checked="" type="checkbox"/> oral <input type="checkbox"/> Rapport <input type="checkbox"/> Exposé <input type="checkbox"/> TP <input checked="" type="checkbox"/>		Written ex. <input checked="" type="checkbox"/> Oral ex. <input type="checkbox"/> Report <input type="checkbox"/> Presentation <input type="checkbox"/> Labworks <input checked="" type="checkbox"/>	
Autre (préciser)		Other (please describe)	

Aides aux étudiants	Student aid

Public ciblé	Student aid

Type de formation	Formation initiale <input checked="" type="checkbox"/> Formation continue <input type="checkbox"/> Apprentissage <input type="checkbox"/> VAE <input checked="" type="checkbox"/>
--------------------------	---

Admission	Admission

Besoins particuliers	Particular needs

Langue(s) utilisée(s) pour l'enseignement : Français	Teaching language : French
Langue(s) utilisée(s) pour le support de cours : Français	Documents language : French
Langue(s) utilisée(s) pour l'évaluation : Français	Assessments language : French

Mots clés : Problèmes de satisfaction de contraintes, programmation par contraintes, modèles SAT, optimisation combinatoire, programmation linéaire en nombres entiers	Keywords : Constraints satisfaction problems, Constraint programming, SAT encoding, combinatorial optimisation, integer linear programming

Bibliographie (auteur, titre, éditeur, année, ISBN)	Bibliography (author, title, publisher, year, ISBN)
R. Dechter, <i>Constraint Processing</i> , Morgan Kaufmann Series in Artificial Intelligence (Elsevier), 2003, ISBN 978-1558608900 F. Rossi, P. Van Beek, T. Walsh, <i>Handbook of Constraint Programming</i> Elsevier, 2006, ISBN: 978-0080463803. K. R. Apt, <i>Principles of Constraint Programming</i> , Cambridge University Press, 2010, ISBN : 978-0521125499 D'autre part, pour les plus curieux, une selection d'articles fondateurs ou de l'état de l'art récent est systématiquement jointe aux supports de cours dans chaque partie.	R. Dechter, <i>Constraint Processing</i> , Morgan Kaufmann Series in Artificial Intelligence (Elsevier), 2003, ISBN 978-1558608900 F. Rossi, P. Van Beek, T. Walsh, <i>Handbook of Constraint Programming</i> Elsevier, 2006, ISBN: 978-0080463803. K. R. Apt, <i>Principles of Constraint Programming</i> , Cambridge University Press, 2010, ISBN : 978-0521125499 Moreover, a detailed selection of seminal and reference papers is joined to course materials given in each part of this course.

¹http://enseignants.insa-toulouse.fr/fr/ameliorer_mon_cours/comment_rediger_les_objectifs_de_son_enseignement.html

² à titre d'exemple, on peut multiplier le présentiel par un facteur fonction du type de pédagogie : 0,9 pour les CM, 0,7 pour les TD, 0,3 pour les TP, 1,5 pour les APP et autres pédagogies actives. Dans tous les cas, cette valeur doit être la plus authentique possible et s'appuyer sur des moyens appropriés pour guider le travail personnel de l'étudiant (exercices non corrigés lors des TD, préparation de TP, exercices « pour aller plus loin », grilles d'auto-évaluation, travail personnel à faire utilisant la bibliographie recommandée, ...)