




5.00 crédits	30.0 h + 15.0 h	Q2
--------------	-----------------	----

Enseignants	Schaus Pierre ;
Langue d'enseignement	Anglais > Facilités pour suivre le cours en français
Lieu du cours	Louvain-la-Neuve
Thèmes abordés	<ul style="list-style-type: none"> • Contraintes et domaine • Aspects pratiques de solvers de contraintes • Problèmes de satisfaction de contraintes (CSP) • Modèles et langages de programmation par contraintes - Méthodes et techniques de résolution de contraintes (consistance, relaxation, optimisation, recherche, intervalles, programmation linéaire, contraintes globales, ...) • Stratégie et technique de recherche • Modélisation et résolution de problèmes (domaines finis et domaines continus) • Applications à la résolution de classes d'applications (par exemple, planification, ordonnancement, allocation de ressources, économie, robotique)
Acquis d'apprentissage	<p>A la fin de cette unité d'enseignement, l'étudiant est capable de :</p> <p>Eu égard au référentiel AA du programme « Master ingénieur civil en informatique », ce cours contribue au développement, à l'acquisition et à l'évaluation des acquis d'apprentissage suivants :</p> <ul style="list-style-type: none"> • AA1.1. Confronté à un problème informatique, il identifie les concepts, algorithmes et structures de données applicables pour le résoudre; et il en tire parti pour décomposer le problème en sous-problèmes et élaborer des méthodes de résolution informatique de ces derniers. • AA1.3. Confronté aux résultats obtenus par le raisonnement et la mise en oeuvre des outils et concepts qu'il a mobilisés, il prend le recul nécessaire pour en vérifier la pertinence, en ce qui concerne les fonctionnalités et la qualité de la solution recherchée. Dans ce contexte, il développera des tests et des vérifications pertinents qui peuvent garantir la qualité de la solution développée. • AA2.4. Dans la phase d'implémentation de la solution, il démontre sa maîtrise des principes, techniques et outils de développement à sa disposition. Il crée un prototype du logiciel à concevoir pour vérifier que le logiciel correspond bien aux attentes du client. Il crée un environnement et une batterie de tests pour s'assurer que la solution développée répond aux fonctionnalités du cahier des charges. En appliquant les techniques de validation et de vérification de programme, il identifie, localise les bugs (bogues) et y remédie. • AA5.4. Il sait se servir d'un ouvrage de référence ou d'un manuel relatif à l'emploi d'un langage informatique ou d'un logiciel, tant en anglais qu'en français. Il comprend un exposé technique fait en anglais. • AA5.5. Au cours du développement d'une application informatique, il en assure la traçabilité et la documentation dans un langage concis et précis : cahier des charges, structure du logiciel et des données qui y sont liées, mode opératoire. Il fait de même quand il s'agit de rédiger un rapport de synthèse décrivant et argumentant les choix (design et technologie) opérés dans le développement d'un projet. <p>Eu égard au référentiel AA du programme « Master [120] en sciences informatiques », ce cours contribue au développement, à l'acquisition et à l'évaluation des acquis d'apprentissage suivants :</p> <ul style="list-style-type: none"> • AA1.démontrer la maîtrise d'un solide corpus de connaissances en informatique, lui permettant de résoudre les problèmes qui relèvent de sa discipline • AA2.1. Analyser le problème à résoudre ou les besoins fonctionnels à rencontrer et formuler le cahier des charges correspondant. • AA2.2. Modéliser le problème et concevoir une ou plusieurs solutions techniques originales répondant à ce cahier des charges. • AA2.4. Implémenter et tester la solution retenue. • AA3.2. Proposer une modélisation et/ou un dispositif expérimental permettant de simuler et de tester des hypothèses relatives au problème étudié dans toute sa complexité. • AA6.5. S'autoévaluer et développer de manière autonome les connaissances nécessaires pour rester compétent dans son domaine. <p>Les étudiants ayant suivi avec fruit ce cours seront capables de:</p> <ul style="list-style-type: none"> • appréhender un ensemble de techniques informatiques permettant de résoudre un problème de satisfaction de contraintes; • développer et étendre un solveur de programmation par contraintes • résoudre une application simple mettant en oeuvre les techniques étudiés; • expliquer les fondements des modèles et des langages de programmation par contraintes; • identifier des classes d'applications où la programmation par contraintes peut être utilisée à bon escient; • modéliser un problème simple sous forme de contraintes et exprimer ce modèle dans un langage de programmation par contraintes y compris l'expression de stratégies de recherche.

	<p>Les étudiants auront développé des compétences méthodologiques et opérationnelles. En particulier, ils auront développé leur capacité à :</p> <ul style="list-style-type: none"> • maîtriser rapidement un nouveau langage de programmation; • utiliser des documents techniques pour approfondir leur connaissance d'un sujet.
Modes d'évaluation des acquis des étudiants	<p>Juin : Pour la première session, la note globale du cours est basée les notes des missions 30%, du projet final 20% de la participation 10% et de l'examen 40%</p> <p> Août : Pour la deuxième session, seul l'examen peut être refait, les autres notes restent inchangées. L'examen d'aout sera soit écrit ou oral. Si le professeur le juge nécessaire, un entretien sur n'importe quel projet pourra aussi être organisé pour vérifier que tous les concepts théoriques sont bien compris. Les projets sont individuels. Cela signifie que tout code source d'un projet jugé copié sur celui d'un autre étudiant entraînera une note de zéro pour l'étudiant, tant pour les projets que pour l'examen. Les mêmes conséquences s'appliqueront à un étudiant qui partage volontairement son code ou le rend disponible à d'autres étudiants. Néanmoins, les étudiants ont la permission d'utiliser des outils d'IA générative pour les aider dans leurs devoirs. Ces outils peuvent fournir de l'inspiration, suggérer des approches de codage ou aider à résoudre des problèmes. Mais : Travail original : Bien que l'IA puisse être un outil, elle ne doit pas être le seul auteur de votre devoir. Votre soumission doit être principalement votre propre travail. Copier et coller directement des solutions issues des sorties de l'IA sans compréhension ni modification est déconseillé. De même, collaborer avec d'autres étudiants est une partie précieuse du processus d'apprentissage, mais copier directement le travail d'un autre étudiant est considéré comme du plagiat. Indication de la source : Chaque fois que vous utilisez l'IA générative pour aider dans votre devoir, vous devez l'indiquer en fournissant un bref commentaire dans votre code sur la manière dont l'IA a été utilisée. Par exemple : <pre># IA utilisée pour suggérer une optimisation pour cette boucle. for i in range(10): ...</pre></p>
Méthodes d'enseignement	<p>La présentation des concepts sera soit proposée sous forme de cours magistraux, de vidéos ou de lecture et sera accompagnée de travaux pratiques (devoirs/micro-projets) sollicitant la mise en œuvre des concepts.</p>
Contenu	<ul style="list-style-type: none"> • La programmation par contraintes: un paradigme de programmation déclarative • Architecture d'un solveur de programmation par contraintes • Contraintes globales et techniques d'implémentation (incrémentalité, etc) • Stratégie et technique de recherche • Modélisation et résolution de problèmes d'optimisation combinatoires à l'aide de la programmation par contraintes • Applications à la résolution de classes d'applications (par exemple, planification, ordonnancement, allocation de ressources, économie, robotique)
Ressources en ligne	<p>https://moodle.uclouvain.be/course/view.php?id=2009 www.minicp.org</p>
Bibliographie	<p>Le site www.minicp.org + lectures suggérées pendant le semestre</p>
Autres infos	<p>Des bases solides en algorithmique et structures de données sont nécessaires pour pouvoir suivre ce cours ainsi qu'une bonne maîtrise de java</p>
Faculté ou entité en charge:	<p>INFO</p>

Programmes / formations proposant cette unité d'enseignement (UE)				
Intitulé du programme	Sigle	Crédits	Prérequis	Acquis d'apprentissage
Master [120] : ingénieur civil en informatique	INFO2M	5		
Master [120] en sciences informatiques	SINF2M	5		
Master [120] : ingénieur civil en science des données	DATE2M	5		
Master [120] en science des données, orientation technologies de l'information	DATI2M	5		