

5.00 crédits	30.0 h + 15.0 h	Q1
--------------	-----------------	----

Enseignants	Pecheur Charles ;
Langue d'enseignement	Anglais
Lieu du cours	Louvain-la-Neuve
Thèmes abordés	<p>Ce cours étudie les principes, les formalismes et les outils utilisés pour modélisation et d'analyse de systèmes informatiques concurrents.</p> <ul style="list-style-type: none"> • Modèles des systèmes concurrents • Sémantique des systèmes concurrents • Propriétés des systèmes concurrents • Vérification de systèmes concurrents
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"> • INFO1.1-3 • INFO2.2-3, INFO2.5 • INFO5.2, INFO5.5 • INFO6.1, INFO6.4 <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"> • SINF1.M1, SINF1.M2 • SINF2.2-3, SINF2.5 • SINF5.2, SINF5.5 • SINF6.1, SINF6.4 <p>1</p> <p>Les étudiants ayant suivi avec fruit ce cours seront capables de</p> <ul style="list-style-type: none"> • maîtriser les concepts et les outils mathématiques qui permettent de modéliser et d'analyser le comportement d'un système informatique concurrent ; • modéliser et un système concret dans un formalisme abstrait approprié ; • Se familiariser aux techniques de vérification automatique utilisées pour l'analyse de ces systèmes. <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"> • modéliser un logiciel existant; • penser à l'aide d'abstractions et utiliser un formalise pour déduire des propriétés d'un système existant .
Modes d'évaluation des acquis des étudiants	<ul style="list-style-type: none"> • 3 missions, 30% de la note finale. • Exercices: test en fin de quadrimestre, 30% de la note finale. • Théorie: examen oral, 40% de la note finale. Une liste de questions est fournie à la fin du cours. <p>Les missions devront être présentées lors du quadrimestre de cours. Ils ne pourront pas être représentées au cours des sessions d'examens ultérieures.</p> <p>Suivant les circonstances, l'examen peut être organisé en distanciel.</p>
Méthodes d'enseignement	<ul style="list-style-type: none"> • Cours magistraux • Séances d'exercices (d'abord des exercices plus théoriques pour acquérir les concepts et ensuite, des séances en salle informatique pour appliquer ces concepts à des systèmes concurrents de plus en plus complexes) • Missions (par groupe de 2 étudiants) <p>Les séances d'exercices et les missions sont "synchronisées" de manière à ce que les exercices préparent les étudiants à réaliser les tâches nécessaires pour mener à bien la mission en cours.</p> <p>Selon les circonstances, tout ou partie des cours et des exercices pourraient être diffusés et enregistrés pour pouvoir être suivis à distance.</p>
Contenu	<ul style="list-style-type: none"> • Modélisation de systèmes concurrents : processus et actions, conditions et choix, concurrence, synchronisation, algèbre de processus.

	<ul style="list-style-type: none"> • Sémantique de systèmes concurrents : machines d'états et systèmes de transitions, traces finies et infinies, parallélisme par entrelacement, équivalences et minimisation. • Propriétés de systèmes concurrents : invariants, propriétés de sûreté et de vivacité, logique temporelle, relations de raffinement. • Vérification de systèmes concurrents : model checking, equivalence checking.
Ressources en ligne	http://icampus.uclouvain.be/claroline/course/index.php?cid=ingi2143
Bibliographie	<p>Livre de référence (obligatoire)</p> <ul style="list-style-type: none"> • J Magee and J Kramer, Concurrency: State Models and Java Programming (2nd Ed.), Wiley, 2006. <p>Autres références</p> <ul style="list-style-type: none"> • H Bowman and R Gomez, Concurrency Theory: Calculi and Automata for Modelling Untimed and Timed Concurrent Systems, Springer, 2006. • AW Roscoe, The Theory and Practice of Concurrency, Prentice Hall, 1998 (http://web.comlab.ox.ac.uk/oucl/work/bill.roscoe/publications/68b.pdf). • E Clarke, O Grumberg and D Peled, Model Checking, MIT Press, 1999. • B Bérard et al., Systems and Software Verification, Springer, 2001.
Faculté ou entité en charge:	INFO

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