



Au vu du contexte sanitaire lié à la propagation du coronavirus, les modalités d'organisation et d'évaluation des unités d'enseignement ont pu, dans différentes situations, être adaptées ; ces éventuelles nouvelles modalités ont été -ou seront- communiquées par les enseignant-es aux étudiant-es.

5 crédits	30.0 h + 15.0 h	Q2
-----------	-----------------	----

Enseignants	Pecheur Charles ;
Langue d'enseignement	Anglais
Lieu du cours	Louvain-la-Neuve
Préalables	Maitrise de la programmation, algorithmique et structures de données (tel que LINFO1121), notions de base en mathématique discrète (tel que LINFO1114 ou LEPL1108), expérience de développement de projets logiciels de taille réduite (tel que LINFO1125).
Thèmes abordés	<ul style="list-style-type: none"> • concepts de qualité du logiciel • spécification et exactitude d'un programme • test de logiciels • vérification et validation de logiciels • révision et audit de logiciels • métriques et meurses de logiciels • fiabilité de logiciels
Acquis d'apprentissage	<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"> • INFO2.3-5 • INFO6.3 <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.M3 • SINF2.3-5 • SINF6.3 <p>Les étudiants ayant suivi ce cours avec fruit seront capables de :</p> <p>1</p> <ul style="list-style-type: none"> • Définir la qualité des logiciels et décrire le rôle des activités d'assurance de la qualité dans le processus de logiciel. • Décrivez comment un contrat peut être utilisé pour spécifier le comportement d'un composant de programme. • Décrire et distinguer les différents types et niveaux de tests (unitaires, d'intégration, les systèmes, et l'acceptation). • Appliquer une variété de stratégies pour les tests et le débogage de programmes simples. • Décrivez comment les outils de vérification statique et dynamique disponibles peuvent être intégrés dans l'environnement de développement de logiciel. • Appliquer des techniques de spécification et d'analyse formelles à laconceptions de programmes avec une faible complexité. • Entreprendre une inspection d'un segment de code de taille moyenne. • Comparer les techniques simples de mesure de logiciels. • Décrire les approches pour estimation de fautes. • Expliquer les problèmes qui existent pour atteindre de niveaux élevés de fiabilité des logiciels. <p>-----</p> <p><i>La contribution de cette UE au développement et à la maîtrise des compétences et acquis du (des) programme(s) est accessible à la fin de cette fiche, dans la partie « Programmes/formations proposant cette unité d'enseignement (UE) ».</i></p>
Modes d'évaluation des acquis des étudiants	<p>En raison de la crise du COVID-19, les informations de cette rubrique sont particulièrement susceptibles d'être modifiées.</p> <p>Le cours inclut les devoirs, comptant pour 40% de la note, et un examen, comptant pour 60% de la note. Les affectations ne peuvent pas être refaites pour la session de septembre. La note reste acquise en septembre.</p>

Ressources en ligne	https://moodleucl.uclouvain.be/course/view.php?id=10913
Bibliographie	<p>Les diapositives de cours ainsi que d'autres informations pertinentes et pratiques relatives au cours seront accessibles sur Moodle. La même plate-forme sera également le moyen de communication entre l'enseignant (s) et les étudiants.</p> <p>Lectures recommandées :</p> <ul style="list-style-type: none"> • Software Quality Engineering: Testing, Quality Assurance, and Quantifiable Improvement. Jeff Tian. 2005, Wiley-IEEE Computer Society Press. • M. Pezzè and Michal Young, Software Testing and Analysis: Process, Principles, and Techniques, Wiley, 2008. • J. Laski, W. Stanley. Software Verification and Analysis. Springer 2009. • N.E. Fenton and S.L. Pfleeger. Software Metrics: A Rigorous and Practical Approach. 2nd edition, Thomson Computer Press, 1996.
Autres infos	<p>Préalables:</p> <ul style="list-style-type: none"> • LINGI1101 : logique mathématique • LSINF1121 : maîtrise de la programmation orienté-objet, de l'algorithmique et des structures de données • LSINF1225 : avoir participé à la réalisation d'un projet logiciel de taille réduite <p>Avoir eu ou avoir simultanément une expérience avec le développement d'un systèmes informatique de taille moyenne à large</p>
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 biomédical	GBIO2M	5		
Master [120] : ingénieur civil en informatique	INFO2M	5		
Master [120] en sciences informatiques	SINF2M	5		