



5.00 crédits

30.0 h + 15.0 h

Q2

Enseignants	Pecheur Charles ;
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"> • 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 mesures de logiciels • fiabilité de logiciels
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"> • 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"> • SIN1.M3 • SIN2.3-5 • SIN6.3 <p>Les étudiants ayant suivi ce cours avec fruit seront capables de :</p> <ol style="list-style-type: none"> 1 <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 la conception 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 l'estimation de fautes. • Expliquer les problèmes qui existent pour atteindre de niveaux élevés de fiabilité des logiciels.
Modes d'évaluation des acquis des étudiants	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.
Méthodes d'enseignement	<ul style="list-style-type: none"> • Cours magistraux hebdomadaires • Séances d'exercices • 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 missions.</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"> • Principes de qualité logicielle • Modèles de logiciel • Test fonctionnel

	<ul style="list-style-type: none"> • Test structurel • Compléments de test • Exécution de tests • Analyse de programmes • Analyse à états finis • Fiabilité logicielle • Mesure de logiciel
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		