






En raison de la crise du COVID-19, les informations ci-dessous sont susceptibles d'être modifiées, notamment celles qui concernent le mode d'enseignement (en présentiel, en distanciel ou sous un format comodal ou hybride).

5 crédits	30.0 h + 30.0 h	Q1
-----------	-----------------	----

Enseignants	Derval Guillaume (supplée Schaus Pierre) ;Schaus Pierre ;
Langue d'enseignement	Français
Lieu du cours	Louvain-la-Neuve
Préalables	<p>Ce cours suppose acquises la maîtrise de la programmation et de la conception de programmes dans un langage orienté-objet tel que Java, la connaissance de structures de données élémentaires et des notions de récursion et de complexité calculatoire telles que visées par le cours LEPL1402.</p> <p><i>Le(s) prérequis de cette Unité d'enseignement (UE) sont précisés à la fin de cette fiche, en regard des programmes/formations qui proposent cette UE.</i></p>
Thèmes abordés	<ul style="list-style-type: none"> • Mesures de complexité d'un algorithme et méthodes d'analyse de complexité. • Algorithmes de tris et recherche dichotomiques. • Structures de données de base (listes, arbres, arbres binaires de recherche) : étude de leurs propriétés abstraites, de leurs représentations concrètes, de leur application et des principaux algorithmes qui les manipulent. • Structures de données avancées (union-find, tables de hachage, tas, arbres binaires équilibrés, représentation et manipulation de graphes, traitement de données textuelles, dictionnaires).
Acquis d'apprentissage	<p>Eu égard au référentiel AA du programme « Bachelier ingénieur civil », 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, AA1.2 • AA2-3.4, AA2-3.5, AA2-3.7 • AA4.2 • AA5.3 <p>Eu égard au référentiel AA du programme « Bachelier 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"> • S1.M1, S1.3 • S2.2, S2.3, S2.4 • S4.3 • S5.4 • S6.1, S6.3 <p>Les étudiants ayant suivi avec fruit ce cours seront capables de</p> <p>1</p> <ul style="list-style-type: none"> • faire un choix argumenté sur l'utilisation des principales structures de données utilisées pour représenter des collections, • utiliser à bon escient les algorithmes existants pour manipuler ces structures de données et analyser leur performance, • concevoir et mettre en oeuvre des variantes des algorithmes étudiés, • tester des algorithmes et des structures de données, • utiliser à bon escient les algorithmes et structures de données documentées dans une l'API • abstraire, modéliser et d'implémenter des solutions efficaces à des problèmes de type « puzzle » algorithmiques. <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"> • analyser de façon critique un problème posé, • tester et debugger des programmes algorithmiques, • implémenter efficacement des algorithmes courts mais non triviaux. • apprendre par eux-mêmes dans un ouvrage de référence et dans la documentation technique complémentaire <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> <ul style="list-style-type: none"> • La participation aux TPs vaut 20% du total. Dans ce total sont intégrées ces différentes facettes: <ul style="list-style-type: none"> • Participation active aux TPs, évaluée par le tuteur • Remise des tâches INGINious dans les temps (essayer raisonnablement est suffisant; vous avez le droit à l'erreur!) <p>La participation est prise en compte également pour la session d'Aout.</p> <ul style="list-style-type: none"> • L'examen est 80% du total. • Un mid-term est organisé aux alentours de S7 sur INGINious. L'évaluation mid-term compte pour 10% du total de l'examen (8% du total global), uniquement si elle fait remonter la moyenne de l'étudiant. La même pondération est appliquée au mois d'aout. La matière est celle des partie 1 à 3.
Méthodes d'enseignement	<p>En raison de la crise du COVID-19, les informations de cette rubrique sont particulièrement susceptibles d'être modifiées.</p> <p>La méthode de pédagogie active suivie dans ce cours est inspirée de l'<i>Approche Par Problèmes (APP)</i> et des classes inversées. Cette méthode repose sur plusieurs <i>phases</i> de travail, certaines encadrées par des <i>tuteurs</i>. Outre les séances tutorées, une des composantes essentielles de cette pédagogie consiste à faire apprendre <i>chaque étudiant par lui-même</i>. La réussite du processus d'apprentissage présuppose donc une implication significative de chaque étudiant. Le rôle du <i>travail en groupe</i> est principalement de débattre des concepts étudiés et, secondairement, d'organiser le travail de chacun. L'apprentissage proprement dit reste de la responsabilité de chaque étudiant.</p> <p>Le travail est organisé en modules que chaque groupe d'étudiants doit accomplir dans un délai strict (typiquement de 2 semaines par module, chaque module contenant deux échéances). Ces modules comportent des <i>questions</i> auxquelles il faut apporter les meilleures réponses possibles (en groupe) et des <i>problèmes de programmation</i>, pour lesquels il faut produire des programmes Java, à faire de manière individuelle.</p>
Contenu	<ul style="list-style-type: none"> • Complexité calculatoire, • Arbres, arbres binaires de recherche, • Arbres équilibrés, • Dictionnaires et tables de hachage, • Files de priorité et tas, • Graphes • Manipulation de données textuelles (pattern matching et de compression)
Ressources en ligne	<p>https://moodleucl.uclouvain.be/course/view.php?id=7682 + Questions sur le site du cours, accessible via Moodle.</p>
Bibliographie	<p>Livre obligatoire: Algorithms, 4th Edition by Robert Sedgwick and Kevin Wayne, Addison-Wesley Professional. ISBN-13: 978-0321573513 ISBN-10: 032157351X</p> <p>Et plus généralement les documents (énoncés des missions, conseils pour l'examen, ...) disponibles sur : http://moodleucl.uclouvain.be/course/view.php?id=7682</p>
Autres infos	<p>Préalables:</p> <ul style="list-style-type: none"> • maîtriser la programmation dans un langage orienté-objet tel que Java • connaître et utiliser correctement de structures de données élémentaires (piles, files, listes, etc.) • avoir des notions en matière de récursion et de complexité calculatoire. <p>Ces préalables sont matières des cours LEPL1401 et LEPL1402.</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] en sciences informatiques	SINF2M	5		
Bachelier en sciences informatiques	SINF1BA	5	LEPL1402	
Bachelier en sciences mathématiques	MATH1BA	5		
Master [60] en sciences informatiques	SINF2M1	5		
Approfondissement en statistique et sciences des données	LSTAT100P	5		
Filière en Informatique	LINFO100P	5		