

5.0 crédits

30.0 h + 30.0 h

1q

Enseignants:	Van Roy Peter ;
Langue d'enseignement:	Français
Lieu du cours	Louvain-la-Neuve
Ressources en ligne:	> http://icampus.uclouvain.be/claroline/course/index.php?cid=LFSAB1402
Préalables :	LFSAB1401 ou LSINF1101
Thèmes abordés :	-- Techniques d'analyse de la complexité calculatoire d'un algorithme ; -- Techniques de raisonnement sur des programmes ; -- Modélisation orientée-objet ; -- Structures de données linéaires et arborescentes ; -- Algorithmes récursifs ; -- Mise en oeuvre de programmes de complexité moyenne ; -- Méthodes de tests et de validation de programmes.
Acquis d'apprentissage	<pre>function showorHide (id) { if (document.getElementById (id).style.display != 'none') { document.getElementById (id).style.display = 'none'; } else { document.getElementById (id).style.display = 'block'; } }</pre> <p>Eu égard au référentiel AA du programme « Bachelier en Sciences de l'Ingénieur, orientation ingénieur civil », ce cours contribue au développement, à l'acquisition et à l'évaluation des acquis d'apprentissage suivants :</p> <p>--</p> <p>Connaissances en sciences fondamentales et polytechniques : AA1.1, AA1.2</p> <p>1.1. Appliquer les concepts, lois, raisonnements à une problématique disciplinaire de complexité cadrée.</p> <p>1.2. Décrire des outils de modélisation et de calcul adéquats pour résoudre une problématique disciplinaire cadrée.</p> <p>--</p> <p>Démarche d'ingénierie : AA2.3, AA2.4, AA2.5, AA2.6, AA2.7</p> <p>2.3. Poser des hypothèses de travail pour la modélisation d'une problématique cadrée.</p> <p>2.4. Modéliser un problème et concevoir une ou plusieurs solutions techniques répondant au cahier des charges.</p> <p>2.5. Implémenter et tester une solution sous la forme d'une maquette, d'un prototype et/ou d'un modèle numérique.</p> <p>2.6. Synthétiser en vue d'expliciter : les hypothèses, la modélisation et la solution proposée.</p> <p>2.7. Porter un regard critique sur des hypothèses prises et sur la pertinence des solutions (autoévaluation individuelle).</p> <p>--</p> <p>Communiquer efficacement oralement et par écrit : AA4.2, AA4.3, AA4.4</p> <p>4.2. Communiquer sous forme graphique et schématique ; interpréter un schéma, présenter les résultats d'un travail, structurer des informations.</p> <p>4.3. Lire, analyser et exploiter des documents techniques (normes, plans, cahier de charge, spécifications...).</p> <p>4.4. Rédiger des documents écrits de synthèse en tenant compte des exigences posées dans le cadre des missions (projets et problèmes).</p> <p>Plus précisément, au terme du cours, l'étudiant sera capable de :</p> <p>--</p> <p>Faire un choix justifié entre plusieurs représentations des informations et plusieurs algorithmes pour les traiter ;</p> <p>--</p> <p>Raisonner sur des (fragments de) programmes : complexité des algorithmes et efficacité des programmes les mettant en oeuvre, raisonnement récursif ;</p> <p>--</p> <p>Appliquer des principes de modélisation orientée-objet ;</p>

	-- Concevoir et appliquer des méthodes de test d'un programme. Acquis d'apprentissage transversaux : -- Analyser un problème de taille moyenne, de proposer une solution informatique pour le résoudre et de la mettre en 'uvre dans un langage de haut niveau. <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>
Modes d'évaluation des acquis des étudiants :	L'évaluation comprend quatre composantes: une interrogation intermédiaire en milieu de quadrimestre, un examen final (écrit) au terme du quadrimestre et un projet obligatoire vers la fin du quadrimestre. Le projet ne peut être fait qu'une fois dans l'année et il compte pour toutes les sessions de l'année académique. La quatrième composante est un bonus donné pour la participation au cours en ligne (MOOC). Le bonus compte pour toutes les sessions de l'année.
Méthodes d'enseignement :	Le cours est donné en ligne comme Massive Open Online Course (MOOC). Le cours magistral complète le cours en ligne et utilise la méthode de la classe inversée. Il y aura aussi des séances de travaux pratiques.
Contenu :	-- Types abstraits de données ; -- Types abstraits linéaires (piles, files, listes, etc. ') et leurs applications ; -- Techniques de représentation des types abstraits linéaires ; -- Modélisation orientée-objet (héritage, composition et réutilisation) ; -- Préconditions, postconditions, invariants ; -- Techniques de raisonnement (règles de déduction, preuves de terminaison, etc.) ; -- Notions de complexité calculatoire ; -- Analyse de la complexité temporelle d'un algorithme ; -- Analyse de la complexité spatiale d'une structure de données ; -- Formulation récursive d'une solution et algorithmes récursifs ; -- Types abstraits arborescents (arbres binaires) et leurs applications ; -- Techniques de représentation des arbres binaires ; -- Mesures de l'efficacité d'un programme ; -- Conception et mise en oeuvre de méthodes de test et de validation.
Bibliographie :	Supports sur le site de cours : -- Dossiers de travail sur les différentes parties du cours Livres de références : -- Peter Van Roy et Seif Haridi, PROGRAMMATION: Concepts, techniques et modèles, Dunod, 2007 -- Peter Van Roy et Seif Haridi, Concepts, Techniques, and Models of Computer Programming, MIT press, 2004
Cycle et année d'étude :	> Bachelier en sciences économiques et de gestion > Bachelier en sciences mathématiques > Bachelier en sciences informatiques > Bachelier en sciences de l'ingénieur, orientation ingénieur civil
Faculté ou entité en charge:	BTCI