

5.00 crédits

30.0 h + 30.0 h

Q2


Cette unité d'enseignement n'est pas accessible aux étudiants d'échange !

Enseignants	Van Roy Peter ;
Langue d'enseignement	Français
Lieu du cours	Charleroi
Préalables	<p>Ce cours suppose acquises des compétences de base en programmation visées par les cours LSINC1101/ LINFO1101 et les notions sur l'algorithmique et les structures de données simples visées par le cours LSINC1402/ 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"> • Paradigmes de programmation (y compris fonctionnel, orienté objet, dataflow et multi-agent) • Sémantique formelle et techniques de raisonnement • Abstraction procédurale (programmation d'ordre supérieur) • Abstraction de données (y compris orienté objet et types de données abstraits) • Programmation concurrente (y compris état partagé, passage de messages et dataflow déterministe) • Programmation symbolique (y compris conception algorithmique)
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 « 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> <ul style="list-style-type: none"> • AA 1.1, 1.2 • AA 2.3, 2.4, 2.5, 2.6, 2.7 • AA 4.2, 4.3, 4.4 <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.I2, S1.I3, S1.I5 • S2.2, S2.3, S2.4 • S5.3, S5.4, S5.5 <p>Les étudiants ayant suivi avec fruit ce cours seront capables de</p> <ul style="list-style-type: none"> • spécifier les problèmes, les diviser dans leurs étapes de base, et concevoir des algorithmes et des abstractions pour les résoudre ; • choisir le bon paradigme de programmation et écrire un programme dans ce paradigme pour résoudre un problème ; • utiliser la sémantique formelle pour expliquer l'exactitude du programme ; • écrire de petits programmes concurrents dans le paradigme du dataflow déterministe. • penser à l'aide d'abstractions (raisonner correctement sur un système qui comprend plusieurs couches d'abstractions, et définir de nouvelles abstractions pour simplifier la résolution d'un problème)
Modes d'évaluation des acquis des étudiants	<ul style="list-style-type: none"> • Test dispensatoire 25% (vers la 7e semaine) • Projet 25% • Examen final (50%) (ou 75% si on refait la partie du test) <p>Le projet est obligatoire et se fait pendant le quadrimestre. Il ne peut être fait qu'une fois et il compte pour toute l'année académique.</p>
Méthodes d'enseignement	<ul style="list-style-type: none"> • Cours magistral chaque semaine • Séances de travaux pratiques en salle informatique chaque semaine, pour résoudre des problèmes simplifiés en utilisant les concepts vu au cours • Un grand projet de conception et d'implémentation pour appliquer ces concepts dans le cadre d'une application plus complexe.
Contenu	<p>Le but de ce cours est d'élargir et approfondir les connaissances en programmation acquises dans les précédents cours. Ce cours aborde les sujets suivants:</p>

	<ul style="list-style-type: none">• Le cours donne un cadre uniforme pour tous les concepts de programmation, organisés en paradigmes de programmation.• Le cours donne une sémantique formelle et des techniques de raisonnement pour tous les paradigmes présentés.• Le cours donne une introduction au lambda calcul comme base de la programmation fonctionnelle et la programmation d'ordre supérieur.• La programmation d'ordre supérieur est utilisée comme principe organisateur pour construire des abstractions procédurales.• La programmation concurrente est présentée en deux formes, le dataflow déterministe et la concurrence à passage de messages.• L'abstraction de données est présentée dans sa forme générale et avec ses deux formes principales, la programmation orientée-objet et les types de données abstraits.• La programmation symbolique et la conception d'algorithmes sont utilisés partout dans le cours.• Cinq paradigmes de programmation importants sont abordés dans ce cours: la programmation fonctionnelle, la programmation orientée objet, la programmation dataflow déterministe, les acteurs dataflow et la programmation avec objets actifs (multi-agent). <p>Des exemples d'applications pratiques sont présentés pour tous les concepts et tous les paradigmes.</p>
Faculté ou entité en charge:	SINC

Programmes / formations proposant cette unité d'enseignement (UE)				
Intitulé du programme	Sigle	Crédits	Prérequis	Acquis d'apprentissage
Bachelier en sciences informatiques	SINC1BA	5	LSINC1101	