






5.00 crédits	30.0 h + 30.0 h	Q2
--------------	-----------------	----

Enseignants	Van Roy Peter ;
Langue d'enseignement	Français
Lieu du cours	Louvain-la-Neuve
Préalables	<p>Ce cours suppose acquises des compétences de base en programmation visées par les cours LINFO1101 ou LEPL1401 et les notions sur l'algorithmique et les structures de données simples 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"> • Paradigmes de programmation : programmation fonctionnelle, programmation orientée-objet et programmation déclarative dataflow ; • Sémantique formelle et techniques de raisonnement sur des programmes ; • Langage noyau et machine abstraite ; • Abstractions de données et modélisation orientée-objet ; • Algorithmes récursifs et programmation avec invariants en exploitant des structures de données linéaires et arborescentes ; • Analyse de la complexité temporelle d'un algorithme et de la complexité spatiale d'une structure de données ; • Non-déterminisme, ordonnancement et équité ; • Mise en oeuvre de programmes de complexité moyenne avec un accent sur les méthodes de tests et de validation de programmes.
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> <p>1</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. Le test dispensatoire (qui est optionnel) et l'examen final seront faits en présentiel ou distanciel, selon les règles en vigueur. Le titulaire se réserve le droit d'interroger un étudiant oralement lors de l'examen final dans certains cas.</p>

Méthodes d'enseignement	<ul style="list-style-type: none"> • Cours magistral chaque semaine (en présentiel ou distanciel, selon les règles en vigueur) • 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 abordera les sujets suivants:</p> <ul style="list-style-type: none"> • Le cours donnera un cadre uniforme pour tous les concepts de programmation, organisés en paradigmes de programmation. • Le cours donnera une sémantique formelle et des techniques de raisonnement pour tous les paradigmes. • La programmation d'ordre supérieur sera utilisée comme principe organisateur pour construire des abstractions procédurales. • La programmation concurrente sera présentée en deux formes, le dataflow déterministe et la concurrence à passage de messages. • L'abstraction de données sera présentée dans sa forme générale et avec ses deux formes principales, la programmation orientée-objet et les types de données abstraits. • La programmation symbolique sera présentée avec la conception d'algorithmes symboliques. • Parmi les paradigmes de programmation abordés dans ce cours seront la programmation fonctionnelle, la programmation orientée objet, la programmation dataflow déterministe et la programmation multi-agent. <p>Des exemples d'applications pratiques seront présentés pour tous les concepts et tous les paradigmes.</p>
Ressources en ligne	LINFO1104 Moodle.
Bibliographie	<p>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</p>
Autres infos	<p>Prérequis:</p> <ul style="list-style-type: none"> • Cours de programmation de seconde année Informatique 2 (LEPL1402) ou équivalent.
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 linguistique	LING2M	5	LINFO1101	
Master [120] : bioingénieur en sciences agronomiques	BIRA2M	5		
Master [120] : bioingénieur en sciences et technologies de l'environnement	BIRE2M	5		
Bachelier en sciences informatiques	SINF1BA	5	LINFO1101	
Filière en Informatique	FILINFO	5		
Master [120] : bioingénieur en chimie et bioindustries	BIRC2M	5		