

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	Van Roy Peter ;
Langue d'enseignement	Anglais
Lieu du cours	Louvain-la-Neuve
Thèmes abordés	<ul style="list-style-type: none"> • paradigmes de programmation concurrente (état partagé, passage de messages et concurrence déclarative) • Concepts majeurs de la programmation qui inclut fonction, objet, classe, abstraction, instantiation, héritage, état, encapsulation, concurrence, dataflow, évaluation paresseuse, non-déterminisme, agent (objet actif), verrou (lock), moniteur, transaction, interblocage (deadlock), programmation d'ordre supérieur, compositionnalité, etc. • Techniques de raisonnement et de conception avec les différents paradigmes pour concevoir des programmes corrects • Applications pratiques dans plusieurs domaines (par exemple, la simulation des circuits logiques, la simulation des ascenseurs, un gestionnaire de transactions).
Acquis d'apprentissage	<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.I5 • S2.2-4 • S5.2, S5.4-5 <p>Les étudiants ayant suivi avec fruit ce cours seront capables de</p> <ol style="list-style-type: none"> 1. définir avec précision et utiliser à bon escient dans les programmes de taille moyenne les concepts-clé de programmation; <ul style="list-style-type: none"> • définir les principaux paradigmes de la programmation concurrente (état partagé, passage de messages et concurrence déclarative), avec les concepts qu'ils contiennent et les propriétés qu'ils donnent aux programmes; • expliquer les relations (similitudes, différences) entre ces différents paradigmes ; • expliquer le lien entre les langages de programmation et les principaux paradigmes de programmation concurrente ; • écrire des programmes de taille moyenne dans ces paradigmes de programmation concurrente. • 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) <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"> • 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.</p>
Méthodes d'enseignement	<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"> • 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 de présenter des concepts avancés des langages de programmation. Entre autres, le cours aborde:</p>

	<ul style="list-style-type: none"> • la programmation fonctionnelle et d'ordre supérieur, la programmation paresseuse, • la programmation concurrente sous ses trois formes principales, à savoir le dataflow déterministe, la concurrence par passage de messages et la concurrence par état partagé, • une présentation rigoureuse du concept de déclarativité, • des techniques avancées de conception d'algorithmes déclaratifs, • la sécurité dans l'abstraction des données, • la conception des systèmes multi-agents y compris dans le langage Erlang, • une présentation rigoureuse de l'état mutatif et son utilité, • les concepts de verrou, moniteur et transactions. <p>Tous ces concepts sont accompagnés d'exercices pratiques.</p>
Ressources en ligne	Moodle LINFO1131.
Bibliographie	Peter Van Roy and Seif Haridi, <i>Concepts, Techniques, and Models of Computer Programming</i> , MIT Press, 2004.
Faculté ou entité en charge:	INFO

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