

La version que vous consultez n'est pas définitive. Cette fiche d'activité peut encore faire l'objet de modifications. La version finale sera disponible le 1er juin.

5.00 crédits	30.0 h + 30.0 h	Q1
--------------	-----------------	----

Langue d'enseignement	Anglais > Facilités pour suivre le cours en français
Lieu du cours	Louvain-la-Neuve
Préalables	Ce cours suppose acquises les concepts, paradigmes et sémantique des langages de programmation visés par le cours LINFO1104
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>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 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; • 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)
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	Le but de ce cours est de présenter des concepts avancés des langages de programmation, en utilisant la programmation concurrente comme principe organisateur. Entre autres, le cours aborde:

	<ul style="list-style-type: none"> • la programmation fonctionnelle, d'ordre supérieur, et 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 avec un accent particulier au langage Erlang et ses concepts, • une présentation rigoureuse de l'état mutatif et son utilité, • les concepts de verrou, moniteur et transaction. <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		