


5.00 crédits

30.0 h + 15.0 h

Q2

Enseignants	Mens Kim ;
Langue d'enseignement	Anglais > Facilités pour suivre le cours en français
Lieu du cours	Louvain-la-Neuve
Thèmes abordés	<p>Au cours de sa carrière, un informaticien ou ingénieur logiciel sera confronté à de nombreux langages et paradigmes de programmation. Pour faire des choix de conception éclairés lors de la sélection d'un langage particulier, il ou elle doit comprendre les principes qui sous-tendent la façon dont les fonctionnalités du langage de programmation sont définis, mis en oeuvre et utilisés.</p> <p>Ce cours examinera, dans une perspective historique, les principes directeurs des grands paradigmes de programmation, à partir des premiers langages de programmation jusqu'aux plus récents. Comme tel, il mettra l'accent sur les grands principes, les points forts et les différences entre divers langages et divers paradigmes de programmation.</p>
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 « Master ingénieur civil en informatique », ce cours contribue au développement, à l'acquisition et à l'évaluation des acquis d'apprentissage suivants :</p> <ul style="list-style-type: none"> • INFO1.2 • INFO2.4-5 • INFO6.3-4 <p>Eu égard au référentiel AA du programme « Master [120] 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"> • • SINF5.2, SINF5.4-5 • SINF6.4 <p>Les étudiants ayant suivi avec fruit ce cours seront capables de</p> <ol style="list-style-type: none"> 1 <ul style="list-style-type: none"> • décrire et différencier les principaux paradigmes de programmation (y compris la programmation procédurale, la programmation fonctionnelle, la programmation logique, la programmation orientée objet, la programmation concurrente, ainsi que les paradigmes de programmation plus récents) • déterminer à quel paradigme appartient un langage de programmation; • identifier et discuter des principes de conception d'un langage ou d'un paradigme donné; • choisir un langage ou un paradigme approprié pour résoudre un problème particulier et argumenter ce choix; • écrire de petits programmes dans une sélection des différents langages et paradigmes vu dans le cours; • placer un langage de programmation par rapport à d'autres dans une perspective historique; • comparer les différents langages de programmation sur base de leurs principes de conception sous-jacents; • comprendre l'impact des différents choix de conception d'un langage (syntaxe, le passage de paramètres, portée, abstraction, ...).
Modes d'évaluation des acquis des étudiants	<p>Tout au long du quadrimestre, en parallèle avec le cours et les labos, les étudiants étudieront en binôme plusieurs langages et paradigmes de programmation vus au cours, en réalisant trois missions de programmation dans trois langages différents, représentatifs de ces paradigmes. Ces missions seront évaluées par des revues de code, des entretiens avec, et des présentations au professeur et assistant(s) du cours. Cette évaluation remplace l'examen traditionnel du cours. Chacune des trois missions comptera pour environ un tiers des points. Si un étudiant a échoué le cours, il devra refaire individuellement en deuxième session toutes les missions de programmation qu'il a échouées.</p>
Méthodes d'enseignement	<p>Le cours consistera des cours magistraux traditionnels au cours desquelles les caractéristiques et les propriétés des différents langages et paradigmes de programmation seront explorés en détail. Les travaux pratiques compléteront ces cours plus théoriques avec des exercices et des missions de programmation plus pratiques dans une sélection de langages de programmation et de paradigmes vus dans le cours théorique.</p>
Contenu	<p>Les étudiants qui terminent ce cours avec succès seront en mesure de:</p>

	<ul style="list-style-type: none"> • décrire et différencier les paradigmes de programmation abordés dans le cours (p.ex. la programmation fonctionnelle, la programmation logique, la réflexion et la métaprogrammation) • déterminer à quel paradigme de programmation appartient un programme ou un langage de programmation donné ; • identifier et discuter des principes de conception d'un langage ou d'un paradigme de programmation donné ; • choisir un langage ou un paradigme adapté à la résolution d'un problème particulier et argumenter ses choix ; • écrire de petits programmes dans les différents langages et paradigmes vus au cours ; • comparer différents langages et paradigmes de programmation du point de vue de leurs principes de conception sous-jacents ; • comprendre l'impact des différents choix de conception de langage.
Ressources en ligne	Les transparents de cours ainsi que d'autres informations pertinentes et pratiques relatives au cours seront accessibles sur Moodle. La même plate-forme sera également le moyen de communication entre l'enseignant et les étudiants.
Bibliographie	<p>References</p> <p>As the programming languages studied in this course may vary from year to year, the recommended references for this course may also vary. Nevertheless, a very useful reference which covers a wide range of programming languages remains: "Principles of Programming Languages - Design, Evaluation and Implementation" by Bruce J. MacLennan.</p> <p>Références</p> <p>Comme les langages étudiés peuvent varier d'une année à une autre, les références conseillées pour ce cours pourront varier également. Néanmoins, une référence très utile qui couvre un large éventail de langages de programmation reste : "Principles of Programming Languages - Design, Evaluation and Implementation" par Bruce J. MacLennan.</p>
Autres infos	<p>Préalable :</p> <ul style="list-style-type: none"> • Avoir un intérêt pour les concepts des langages de programmation, comme par exemple vu dans les cours LINFO1104 "Concepts des langages de programmation" et LINFO1131 "Concepts de programmation concurrente". • Plus l'étudiant a été confronté à différents langages de programmation, plus il ou elle appréciera ce cours.
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] : ingénieur civil en informatique	INFO2M	5		
Master [120] en sciences informatiques	SINF2M	5		