

6.0 crédits	30.0 h + 30.0 h	1q
-------------	-----------------	----

Enseignants:	Bonaventure Olivier ; Pecheur Charles ;
Langue d'enseignement:	Français
Lieu du cours	Louvain-la-Neuve
Thèmes abordés :	<p>p { margin-bottom: 0.08in; } -- Principes de base de la programmation -- Valeurs, variables, affectation -- Types primitifs, représentation des nombres -- Expressions et instructions, conditions et boucles -- Méthodes, paramètres, résultats et appels -- Spécifications -- Edition, compilation et exécution de programmes -- Programmation orientée-objets -- Classes, objets, instances -- Variables de classe et d'instance -- Visibilité, portée et durée de vie d'une variable -- Interfaces et implémentations -- Extension d'une classe et héritage, polymorphisme -- Fonctionnalités et structures de données -- Chaînes de caractères -- Fichiers et entrées-sorties -- Mécanismes d'exception -- Tableaux simples et tableaux multidimensionnels -- Utilisation de packages -- Structures chaînées -- Méthodologie de la programmation -- Analyse d'un problème informatique -- conception, spécification et mise en oeuvre d'une solution</p>
Acquis d'apprentissage	<p>p { margin-bottom: 0.08in; } p { margin-bottom: 0.08in; } Les étudiants ayant suivi avec fruit ce cours seront capables de -- démontrer une bonne compréhension des concepts et de la méthodologie de la programmation orientée-objet,</p>

	<p>-- utiliser à bon escient les éléments d'un langage orienté-objet tel que Java, -- analyser un problème simple, proposer une solution informatique pour le résoudre et la mettre en oeuvre en Java. Les étudiants auront développé des compétences méthodologiques et opérationnelles. En particulier, ils auront développé leur capacité à : -- décomposer un problème simple en sous-problèmes; -- exprimer la résolution d'un problème sous forme d'algorithme; -- définir et spécifier avec rigueur et précision les fonctionnalités attendues d'un programme informatique; -- lire et comprendre des programmes existants; -- utiliser des outils informatiques de support à la programmation; -- exploiter la documentation technique d'un langage informatique; -- réaliser une application informatique de petite taille sous forme de programme Java de bonne qualité; -- identifier les cas à tester pour la validation d'un programme simple.</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>
<p>Modes d'évaluation des acquis des étudiants :</p>	<p>-- Examen écrit -- Les projets et travaux réalisés durant le quadrimestre sont évalués contribuent à a cote finale -- Un test est organisé en 4e ou 5e semaine pour que les étudiants (en première année à l'université) aient un moyen de valider rapidement leur méthode de travail, leur acquisition des connaissances et le niveau d'exigence à l'université</p>
<p>Méthodes d'enseignement :</p>	<p>L'enseignement est organisé de manière traditionnelle avec un cours magistral complété de séances d'exercices et de travaux à rendre. Les travaux pratiques consistent d'une part en séances en salles de cours où des exercices simples sont résolus avec l'aide d'un assistant et, d'autre part, en trois ou quatre projets de taille réduite résolus par groupes de 2 étudiants.</p>
<p>Contenu :</p>	<p>Le premier chapitre concerne la construction d'algorithmes. On donne aux étudiants des objets de base munis d'actions primitives. On montre ensuite comment construire des algorithmes arbitrairement complexes en utilisant les compositions séquentielle, conditionnelle et répétitive et en décomposant systématiquement les problèmes en sous-problèmes. On insiste sur le fait qu'un code correct doit être construit à partir d'assertions intermédiaires. Dans ce chapitre, le langage Java est utilisé de manière limitée et contrôlée pour traduire les algorithmes conçus de façon théorique afin de les faire exécuter par un ordinateur.</p> <p>Le deuxième chapitre aborde la façon dont les données sont encodées en binaire. On étudie en détail les systèmes de numérations et les algorithmes de conversion d'un système à l'autre. On définit aussi le mode de représentation des entiers en complément à deux et la représentation des nombres en virgule flottante.</p> <p>Le troisième chapitre décrit le fonctionnement de l'unité centrale d'un ordinateur fictif simple. Le langage machine de cette unité centrale est défini avec précision et les techniques de programmation en langage machine sont expliquées.</p> <p>Le quatrième chapitre est une introduction aux langages de programmation, illustrée avec le langage Java. On ne cherche pas à donner un aperçu global du langage Java mais à définir de manière vraiment précise un ensemble significatif de notions de base (suffisant pour écrire des algorithmes simples). On étudie en détails les notions suivantes : types de données primitifs, littéraux, variables, tableaux, la classe String, méthodes statiques, expressions, un sous-ensemble simple des instructions composées, conversions entre types primitifs et surcharge des noms de méthodes. Avec ces éléments du langage Java, on construit de manière rigoureuse des algorithmes classiques : recherche en tables, tris de tableaux.</p>
<p>Bibliographie :</p>	<p>-- transparents disponibles en ligne</p>
<p>Cycle et année d'étude :</p>	<p>> Bachelier en sciences mathématiques > Master [120] en linguistique > Bachelier en sciences informatiques > Bachelier en sciences de l'ingénieur, orientation ingénieur civil architecte > Bachelier en sciences économiques et de gestion > Master [120] en sciences et technologies de l'information et de la communication</p>
<p>Faculté ou entité en charge:</p>	<p>INFO</p>