Directeur(trice): Nadia Ghazzali
CPPC - Mathématiques et informatique
819 376-5011, poste 3802

Bureau du registraire
1 800 365-0922 ou 819 376-5045
www.uqtr.ca

Structure du programme et liste des cours

Certificat en informatique

(Cheminement: 1)

À moins d'indication contraire, un cours comporte trois (3) crédits

Cours optionnels (30 crédits)

L'étudiant doit suivre dix cours parmi la liste des cours suivants (trente crédits) :

ASY1009 - Outils de développement Web

Initier l'étudiant à la création et à la publication de documents Web; rendre l'étudiant apte à utiliser des produits commerciaux de création de sites Web.

Initiation à l'Internet et au Web; introduction au langage HTML. Etapes de la conception d'un site Web; publication d'un site Web, utilisations de types différents d'outils de conception de site Web : éditeur de pages HTML (tels que webExpert, Netscape Composer, EditPlus, HomeSite...); éditeur de sites Web (tels que Dream Weaver (Macromedia), Golive (Adobe), FrontPage (Microsoft)...); éditeur d'images sous format « bitmap » ou vectoriel (tels que Photoshop Elements (Adobe), Fireworks (Macromedia), Freehand (Macromedia)...); logiciel de création d'animation (tels que Flash (Macromedia), LiveMotion (Adobe)...).

Environnement : Les logiciels de chaque type d'outils qui seront utilisés seront annoncés lorsque le cours sera mis à l'horaire.

GCO1006 - Communication graphique

Ce cours vise à familiariser l'étudiant au langage graphique et à l'utilisation de moyens complexes d'exploitation du graphisme et de l'image pour des fins de communication sociale.

Apprentissage du langage graphique : espace, direction, ligne, surface, contraste, etc. Composition graphique. Etudes des rapports entre le texte et l'image et de la mise en page. Etudes de divers aspects liés à la couleur. Animation d'objets graphiques. Utilisation de logiciels d'application pour le traitement graphique.

Note : Les étudiants doivent avoir, pour suivre ce cours, les logiciels de la suite Adobe suivants: Illustrator et Photoshop.

GIF1003 - Réseaux, télématique et technologies informatiques

Connaître et appliquer les différents concepts reliés aux nouvelles technologies de l'information et des communications. Fournir les bases de fonctionnement des réseaux: architecture en couches et protocoles. Présenter les aspects liés à la communication des réseaux locaux, les concepts du réseau Internet et le modèle client/server. Se familiariser avec les concepts de la sécurité et leurs impacts sur les nouvelles technologies informatiques.

INF1001 - Programmation Web

Permettre à l'étudiant de s'initier aux concepts du Web. A travers des exemples détaillés, l'étudiant apprend à créer et à publier des sites Web en exploitant les nouvelles technologies du Web. Il se familiarise avec les langages de programmation propres au Web.

Principaux éléments du Web; langage de balisage, conception de page Web, validation, organisation et gestion d'un site Web, programmation coté client

et coté serveur, introduction à XML.

Technologies : HTML, CSS, DOM, JavaScript, php.

Ce cours utilise les langages HTML, JavaScript, VbScript, XML et l'outil de conception de pages Web EditPlus.

INF1002 - Introduction à la programmation objet

Permettre à l'étudiant de se familiariser avec l'informatique comme discipline et surtout à la programmation impérative et orientée objet. Présentation des structures de commandes, des types, des objets, des structures de données élémentaires (tableau, chaîne de caractères); mécanismes de structuration d'applications : sous-programmes, notions de classe, d'objet, d'application; développement d'applications : choix des données et des algorithmes, codification, compilation, exécution, mise au point de classes et d'applications.

Introduction à l'informatique; outils de construction d'application : syntaxe et sémantique d'un langage de haut niveau; variables, types, expressions, affectation; entrée-sortie élémentaire; structures de sélection et d'itération : sous-programmes et passage de paramètres : introduction au paradigme objet : notions de classe, d'objet, d'état et de méthode, d'héritage; algorithmes et résolution de problèmes (problem-solving) : stratégies de solution de problèmes, rôles de l'algorithme dans la résolution de problèmes; stratégies d'implantation; structure de données de base : tableaux, chaînes de caractères; stratégie de vérification : conception de points-tests.

Ce cours utilise le langage de programmation Java sur la plate-forme Eclipse. Il comporte 18 heures d'atelier.

INF1004 - Structures de données et algorithmes

Permettre à l'étudiant de se familiariser avec les structures de données classiques et les algorithmes qui leur sont associés; réaliser des implantations statiques et dynamiques de ces structures : faire l'évaluation de la complexité spatiale et temporelle dans les cas simples; étudier la récursion et la comparer avec l'itération.

Revue des concepts élémentaires de programmation; bases de la programmation Objet: encapsulation, dissimulation de l'information; séparation du comportement et de l'implantation; héritage et polymorphisme; conception par héritage et par composition; utilisation de fichiers; les principales structures de données: liste, pile, file, table d'adressage dispersé, arbre, graphe; implantation statique et dynamique; les algorithmes de fouille, de tri, les fonctions de hachage et les stratégies de traitement des collisions, parcours d'arbres et de graphes; le concept de récursion : les fonctions mathématiques récursives; comparaison avec les fonctions itératives correspondantes; implantation de récursion à l'aide de piles; analyse élémentaire de la complexité des algorithmes: complexité spatiale et complexité temporelle; notation "grand O", comportement du meilleur cas, du cas moyen et du pire cas; principales classes de complexité d'algorithmes; stratégies de test pour les classes et les applications.

Ce cours utilise le langage de programmation Java et la plateforme Eclipse.

INF1006 - Analyse et modélisation

Familiariser l'étudiant avec le processus de développement de systèmes informatiques, et plus particulièrement avec les phases initiales. L'initier à l'exercice de l'analyse des besoins, à la spécification et à la modélisation des systèmes ainsi qu'à l'évaluation de ces étapes. Explorer les fondements et l'évolution des méthodes d'analyse.

Processus de développement et activités initiales: analyse des besoins et spécifications; importance de l'analyse dans le processus de développement; étude détaillée et application d'une méthode (UML); différentes étapes du processus: analyse des besoins, formalisation et validation; analyse et modélisation orientées objet; outils GLAO (CASE) de modélisation; dossier des spécifications et des exigences d'un logiciel; normes, critères de qualité de la spécification; revue technique; réalisation d'un dossier de spécifications.

Le cours comporte 12 heures d'atelier.

INF1007 - Conception de logiciels

Initier l'étudiant au processus de conception et le sensibiliser à ses difficultés et à son importance. Procéder à l'étude détaillée et à l'application d'une méthode. Permettre à l'étudiant d'élaborer des solutions de qualité.

Problématique du processus de conception. Importance de la conception dans le processus de développement. Processus de conception : principes, méthode et notation. Étude détaillée et application d'une méthode. Différentes étapes du processus. Conception et styles d'architectures logicielles. Patrons

de conception (GRASP). Qualité de la conception : élaboration de solutions réutilisables, maintenables, extensibles, etc. Conception orientée objet. Caractéristiques d'une bonne conception détaillée (modularité, abstraction, cohésion, couplage, etc.). Outils d'aide à la conception. Documentation de la conception. Revue de la conception. Réalisation d'un projet.

INF1009 - Réseaux d'ordinateurs I

Le but du cours est de fournir les bases indispensables à la compréhension des architectures des réseaux informatiques qui permettent l'échange des données entre ordinateurs. Il précise le fonctionnement des réseaux : architecture en couches, protocoles.

Étant donné que la connaissance du domaine est compliquée par la richesse du vocabulaire, on présente progressivement les notions et le vocabulaire à l'aide d'exemples simples. Une première partie présente les aspects liés à la transmission des informations : circuits de données, protocoles de liaison, réseaux à circuits virtuels. Une fois les principes présentés, on détaille le formalisme des sept couches OSI : concepts, description du modèle. Une troisième partie est consacrée aux réseaux locaux (Ethernet, Token Ring, Sans fil). Finalement, on explique les principes des réseaux IP.

INF1017 - Gestion de parcs d'ordinateurs

Familiariser l'étudiant aux méthodes de gestion de parcs d'ordinateurs et à la gestion courante d'un système informatique : installation et gestion des serveurs, des périphériques, des comptes usagers. Gestion de réseaux, gestion de la mémoire. Gestion des incidents, des licences, de la sécurité. Gestion du câblage. Optimisation des performances.

Installation de systèmes d'exploitation courants (LINUX, Windows); gestion courante d'un parc informatique : comptes, partage d'imprimante, etc.; installation de passerelles matérielles et logicielles entre différents systèmes d'exploitation, partage de fichiers, archivage, sauvegarde.

Les étudiants font l'installation et la gestion de serveurs et la gestion de systèmes informatiques.

INF1030 - Introduction au développement d'applications mobiles (version Android)

Dans ce cours, l'étudiant sera amené à comprendre le cycle de développement d'applications mobiles. Utilisant le langage Java, les étudiants pourront créer des programmes tant standalone que déployés sur des plateformes mobiles. L'emphase sera mise sur les apprentissages des meilleurs pratiques de développement d'application mobiles et ce dans le but de faciliter et d'augmenter l'efficacité de ce développement. Les étudiants pourront aussi comprendre le fonctionnement de la machine virtuelle Dalvik comme plateforme pour le développement d'applications Android.

INF1032 - Introduction au développement d'applications mobiles (plateforme IOS)

Dans ce cours, l'étudiant sera amené à comprendre le cycle de développement d'applications mobile. Utilisant le langage Objective C, les étudiants pourront créer des programmes tant standalone que déployés sur des plateformes mobiles. L'emphase sera mise sur les apprentissages des meilleurs pratiques de développement d'application mobiles et ce dans le but de faciliter et d'augmenter l'efficacité de ce développement. Les étudiants pourront aussi apprendre comment développer des applications sur des plateformes mobiles IOS (ex : IPhone et IPad).

INF1034 - Introduction aux interfaces utilisateur

Amener l'étudiant à comprendre le comportement humain dans son interaction avec un environnement logiciel interactif, à développer une interface utilisateur simple et à évaluer l'ergonomie d'un logiciel interactif.

Concepts de base de l'ergonomie du logiciel et des interactions personne-machine : modèles, principes et lignes directrices; introduction à la conception d'interfaces utilisateur : processus, outils, méthodologie et normes; outils de développement d'interfaces graphiques : utilisation des contrôles et des objets prédéfinis, propriétés, méthodes et gestion des événements; évaluation des interfaces : test, enquête et expérimentation.

Ce cours comporte des exercices en Java.

INF1035 - Concepts avancés en objet

Amener l'étudiant à comprendre les mécanismes du développement orienté-objet. Maîtriser les concepts relatifs au paradigme objet. Utiliser un environnement de développement (IDE). Utiliser le paradigme de programmation événementielle. Utiliser les principes (et mécanismes) de la programmation par contrat. Intégrer la programmation aspect.

Etude des notions importantes de la programmation orientée-objet : héritage simple et multiple, hiérarchie des classes, polymorphisme, notion de liaison statique et de liaison dynamique, etc. Notion d'interface et de classe abstraite. Notion d'extension de classes. Généricité. Pattern de conception (GoF). Refactoring. Notions de robustesse et de sécurité dans les applications. Programmation événementielle : événements, messages, gestion des exceptions. Programmation par contrats. Programmation aspect : notions d'aspect, de poincut, de point de jointure et d'advice.

Les langages utilisés dans ce cours sont C#, Java et AspectJ (à titre comparatif). Les environnements de développement sont les plateformes Eclipse et Visuel C#. Le cours comporte 18 heures d'atelier.

PRO1026 - Programmation et enseignement

Développer la rigueur scientifique par une étude systématique de la programmation. Acquérir les bases d'un langage structuré de programmation.

Familiarisation avec les outils mathématiques permettant de résoudre des problèmes classiques avec les différentes classes de solutions informatiques et avec la vérification de programmes. Perspective historique de la programmation et de l'informatique. Structure d'un ordinateur. Différentes stratégies de résolutions de problèmes concrets. Etude d'un langage de programmation. Programmation de problèmes faisant appel à différentes classes de solutions : itération, récursion. Outils formels pour la réalisation de programmes : logique, mathématique, récurrence. Eléments de vérification de programmes.

PRO1027 - Programmation scientifique en C

Approfondir les concepts fondamentaux de la conception et de la production de logiciel. Développer l'habileté à réaliser des logiciels de qualité. Réaliser des applications pratiques dans le domaine des méthodes numériques.

Étude des langages C et C++. Qualité des programmes : clarté, lisibilité, modifiabilité, efficacité, facilité d'usage, robustesse. Programmation modulaire. Programmation en langage C ou C++. A titre indicatif, les applications sont choisies parmi les suivantes : statistiques, manipulation de matrices, résolution de systèmes d'équations linéaires, interpolation, approximation, dérivation et intégration numériques.

SDD1001 - Introduction à la science des données

Ce cours aborde les langages modernes utilisés en sciences des données comme Python, les bibliothèques de statistiques, de calcul symbolique ainsi que leurs applications. Les langages seront abordés dans les paradigmes orientés objets et fonctionnels.

SDD1002 - Modélisation et simulation

Ce cours aborde les différentes techniques théoriques et pratiques de simulation, de modélisation, et de visualisation, des données. L'étudiant sera amené à explorer plusieurs outils utilisés dans les sciences des données. Ce cours aborde aussi les notions liées à la généralisation: sous-ajustement, sur-ajustement, validation croisée. Approches: classification, régression, agrégation (clustering). Principaux éléments: Outils de simulation, de visualisation et manipulation de données comme Python ou R ou Orange, ou Tableau, autre.

SDD1003 - Gestion des entrepôts de données

Les bases de données non relationnelles n'incorporent pas le modèle table/clé que les systèmes de gestion de base de données relationnelles promeuvent. Ce cours permet à l'étudiant de comprendre, de modéliser et de manipuler les données de masse à l'aide des nouvelles technologies de Bases de données (BD) non relationnelles. Aussi, l'étudiant sera introduit à comment lire et manipuler les données des bases de données non relationnelles dans des applications web. L'architecture des données et l'optimisation des requêtes seront également abordées.

Les exemples de base de données abordées sont les Bases de données documents et les Bases de données clé-valeur.

SIF1015 - Systèmes d'exploitation

L'étudiant se familiarise avec les concepts fondamentaux des systèmes d'exploitation : structure générale d'un système d'exploitation, gestion des différents sous-systèmes : processus/thread, communication inter-processus/thread, systèmes de fichiers, accès réseau, gestion de la mémoire.

Gestion des processus et des threads; gestion des entrées/sorties tels que les dispositifs Windows; communication inter-processus : tubes anonymes et

nommés, files de messages, sockets, signaux; introduction aux principes de fonctionnement d'un système de fichiers; introduction aux concepts de la gestion de la mémoire; introduction à la communication réseau par sockets.

Les étudiants expérimentent les concepts théoriques par des projets de développement d'utilitaires dans un environnement de développement ouvert sous une plateforme telle que UNIX/LINUX/ANDROID.

Règlement pédagogique particulier : Pour les étudiants du baccalauréat en génie électrique (concentration génie informatique) (7144) le préalable est GEI1072 Résolution de problèmes d'ingénierie en C. Pour les étudiants du baccalauréat en informatique (7833) les préalables sont INF1004 Structures de données et algorithmes et SIF1053 Architecture des ordinateurs.

SIF1053 - Architecture des ordinateurs

L'étudiant prendra connaissance des éléments fondamentaux des architectures et des composantes des ordinateurs modernes tant statiques que mobiles (ex : architectures ARM, X86, x86-64), des organes périphériques.

Etude des composantes des ordinateurs modernes: mémoire, circuits et représentation de l'information. Unité de commande, représentation des instructions machine, notions d'adressage, notions de microprogrammation et machine virtuelle. Unité arithmétique et logiques. Introduction des notions de communication série et parallèle.

SMI1001 - Bases de données I

Le cours vise à donner une solide introduction tant théorique que pratique aux concepts informatiques fondamentaux ayant trait aux bases de données, particulièrement aux bases de données relationnelles. Outre la modélisation des données qui constitue une partie importante de ce cours, l'apprentissage du langage SQL et d'un SGBD (Système de Gestion de Bases de Données) moderne seront également des objectifs importants de ce cours.

Introduction aux bases de données : modèles et langages, les différents utilisateurs et leurs besoins, structure et architecture des SGBD. Le modèle Entité-Relation, modélisation UML. Le modèle relationnel : aperçu de l'algèbre et du calcul relationnel, notion de vue. Le langage SQL, aperçu d'autres langages relationnels. Les contraintes d'intégrité, les données manquantes et les valeurs nulles. La conception des bases de données relationnelles, dépendances fonctionnelles, dépendances multivaluées, dépendances de jointure, normalisation, formes normales 1FN, 2FN, 3FN, FNBC, 4FN et 5FN. Autres types de bases de données : relationnelles-objets et multidimensionnelles. Introduction aux données multimédias et XML.

Ce cours utilise le système de gestion de bases de données Oracle, le langage PL/SQL et l'utilitaire TOAD. Le cours comporte 18 heures d'atelier.