Baccalauréat en informatique (cheminement DEC-BAC)

7533

Personne ressource: Valérie Noël

819 376-5011, poste 3802

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

Grade: Bachelier ès sciences appliquées (B.Sc.A.)

Crédits: 90

Présentation

En bref
Le programme de baccalauréat en informatique de l'Université du Québec à Trois-Rivières est un programme d'informatique fondamentale axé principalement sur la conception et la production de logiciels.

Le programme est organisé de façon à ce que l'étudiant acquière les habiletés et les notions fondamentales d'informatique dans les cours obligatoires. Un bon éventail de cours optionnels répartis dans trois cheminements permet d'abord à l'étudiant de pouvoir se spécialiser soit dans le développement de logiciels (01) ou soit le développement d'applications Web et mobiles (02) ou soit les sciences des données (03) et aussi de se familiariser avec des sujets plus avancés comme le graphisme, le traitement d'image, l'intelligence artificielle, les architectures réparties.

Concentrations, profils, cheminements
Au travers de son choix de cours optionnels, l'étudiant pourra donc se spécialiser soit en développement de logiciels (cheminement 01) ou en développement d'applications Web et mobiles (cheminement 02) ou en sciences des données (cheminement 03).

IMPORTANT : Dans sa demande d'admission, le candidat doit préciser le cheminement de son choix.

Atouts UQTR
Les étudiantes et étudiants québécois inscrits à temps plein dans ce programme sont admissibles à un programme de bourses incitatives: Programme de bourses Perspective Québec.

Admission

Trimestre d'admission et rythme des études

Automne, hiver.

Ce programme est offert à temps complet et à temps partiel.

Conditions d'admission

Études au Québec
Base DEC

L'étudiant admis au programme Techniques en informatique (420.A0) au Cégep de Trois-Rivières, au Cégep de Drummondville, au Cégep de Shawinigan ou au Cégep de Sorel-Tracy peut intégrer le DEC-BAC en informatique.

Les cours de mathématiques de niveau collégial 201-NYA-05 Calcul différentiel et 201-NYB05 Calcul intégral (ou leurs équivalents sont préalables à l'admission au programme de baccalauréat en informatique.

Études hors Québec

Ce programme n'est pas offert aux candidats de l'international.

Structure du programme et liste des cours

Développement de logiciels

(Cheminement: 1)
À moins d'indication contraire, un cours comporte trois (3) crédits.

Cours obligatoires (60 crédits)

INF1001	Programmation Web
INF1002	Introduction à la programmation objet
INF1004	Structures de données et algorithmes (INF1002)
INF1006	Analyse et modélisation (INF1004)
INF1007	Conception de logiciels (INF1006)
INF1008	Analyse et conception d'algorithmes (ALG1001 ou PIF1005; INF1004)
INF1009	Réseaux d'ordinateurs I (SIF1015)
INF1010	Réseaux d'ordinateurs II (INF1009)
INF1014	Aspects juridiques de l'informatique (1 crédit)
INF1015	Stage d'informatique I
INF1016	Stage d'informatique II
INF1034	Introduction aux interfaces utilisateur (INF1002)
INF1035	Concepts avancés en objet (INF1004)
PIF1005	Mathématiques pour informaticiens I
PIF1006	Mathématiques pour informaticiens II (PIF1005)
SIF1015	Systèmes d'exploitation
SIF1053	Architecture des ordinateurs
SMI1001	Bases de données I
SMI1002	Bases de données II (SMI1001)
STT1001	Probabilités et statistiques
TIN1003	Science, technologie et société (2 crédits)

Cours optionnels (21 crédits)

L’étudiant suit le cours suivant (3 crédits) :

INF1011	Génie logiciel (INF1007)

L’étudiant doit suivre 9 crédits de cours informatique :

INF1012	Élaboration et gestion de projets scientifiques (INF1011)
INF1013	Environnement de développement et composantes logicielles (INF1005 ou INF1035; SIF1015)
INF1018	Analyse de programmes (INF1035)

L'étudiant doit suivre six (6) ou neuf (9) crédits parmi les cours d'informatique suivants :

IAR1001	Intelligence artificielle (STT1001 ou STT1003; INF1008; INF1004)
INF1036	Sujets spéciaux en TI
PRO1027	Programmation scientifique en C (INF1002 ou PRO1026)
SIF1033	Traitement d'image (INF1004; STT1001 ou STT1042)

L’étudiant peut suivre de 0 à 3 crédits parmi les cours suivants :

INF1020	Commerce électronique
INF1030	Introduction au développement d'applications mobiles (version Android) (INF1005 ou INF1035)
INF1031	Développement d'applications mobiles avancées (version Android) (INF1030)

Cours complémentaires (9 crédits)
L'étudiant choisit neuf (9) crédits de cours complémentaires parmi les suivants. Certains cours sont offerts en ligne. Consulter la liste.

Développement d'applications Web et mobiles

(Cheminement: 2)
À moins d'indication contraire, un cours comporte trois (3) crédits.

Cours obligatoires (60 crédits)

INF1001	Programmation Web
INF1002	Introduction à la programmation objet
INF1004	Structures de données et algorithmes (INF1002)
INF1006	Analyse et modélisation (INF1004)
INF1007	Conception de logiciels (INF1006)
INF1008	Analyse et conception d'algorithmes (ALG1001 ou PIF1005; INF1004)
INF1009	Réseaux d'ordinateurs I (SIF1015)
INF1010	Réseaux d'ordinateurs II (INF1009)
INF1014	Aspects juridiques de l'informatique (1 crédit)
INF1015	Stage d'informatique I
INF1016	Stage d'informatique II
INF1034	Introduction aux interfaces utilisateur (INF1002)
INF1035	Concepts avancés en objet (INF1004)
PIF1005	Mathématiques pour informaticiens I
PIF1006	Mathématiques pour informaticiens II (PIF1005)
SIF1015	Systèmes d'exploitation
SIF1053	Architecture des ordinateurs
SMI1001	Bases de données I
SMI1002	Bases de données II (SMI1001)
STT1001	Probabilités et statistiques
TIN1003	Science, technologie et société (2 crédits)

Cours optionnels (21 crédits)

L'étudiant suit le cours suivant (3 crédits) :

INF1011	Génie logiciel (INF1007)

L'étudiant doit suivre 9 crédits de cours informatique suivants :

INF1020	Commerce électronique
INF1030	Introduction au développement d'applications mobiles (version Android) (INF1005 ou INF1035)
INF1031	Développement d'applications mobiles avancées (version Android) (INF1030)

L'étudiant doit suivre six (6) ou neuf (9) crédits parmi les cours d'informatique suivants :

INF1021	Les architectures réparties (INF1009)
INF1032	Introduction au développement d'applications mobiles (plateforme IOS) (INF1005 ou INF1035)
INF1036	Sujets spéciaux en TI
SIF1032	Graphisme par ordinateur (MAP1006 ou PIF1006; INF1004)

L'étudiant peut suivre zéro (0) ou trois (3) crédits parmi les cours d'informatique du bloc de cours optionnels du cheminement

développement de logiciels.

INF1012	Élaboration et gestion de projets scientifiques (INF1011)
INF1013	Environnement de développement et composantes logicielles (INF1005 ou INF1035; SIF1015)
INF1018	Analyse de programmes (INF1035)
PRO1027	Programmation scientifique en C (INF1002 ou PRO1026)

Cours complémentaires (9 crédits)
L'étudiant choisit neuf (9) crédits de cours complémentaires parmi les suivants. Certains cours sont offerts en ligne. Consulter la liste.

Science des données

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

Cours obligatoires (60 crédits)

INF1001	Programmation Web
INF1002	Introduction à la programmation objet
INF1004	Structures de données et algorithmes (INF1002)
INF1006	Analyse et modélisation (INF1004)
INF1007	Conception de logiciels (INF1006)
INF1008	Analyse et conception d'algorithmes (ALG1001 ou PIF1005; INF1004)
INF1009	Réseaux d'ordinateurs I (SIF1015)
INF1010	Réseaux d'ordinateurs II (INF1009)
INF1014	Aspects juridiques de l'informatique (1 crédit)
INF1015	Stage d'informatique I
INF1016	Stage d'informatique II
INF1034	Introduction aux interfaces utilisateur (INF1002)
INF1035	Concepts avancés en objet (INF1004)
PIF1005	Mathématiques pour informaticiens I
PIF1006	Mathématiques pour informaticiens II (PIF1005)
SIF1015	Systèmes d'exploitation
SIF1053	Architecture des ordinateurs
SMI1001	Bases de données I
SMI1002	Bases de données II (SMI1001)
STT1001	Probabilités et statistiques
TIN1003	Science, technologie et société (2 crédits)

Cours optionnels (21 crédits)

L'étudiant suit les cours suivants (12 crédits) :

SDD1001	Introduction à la science des données
SDD1002	Modélisation et simulation (STT1001; PIF1005)
SDD1003	Gestion des entrepôts de données (SMI1002; INF1004)
SDD1004	Apprentissage Automatique et applications (SDD1002)

L'étudiant choisit 9 crédits parmi les cours suivants :

INF1011	Génie logiciel (INF1007)
INF1012	Élaboration et gestion de projets scientifiques (INF1011)
INF1013	Environnement de développement et composantes logicielles (INF1005 ou INF1035; SIF1015)
INF1018	Analyse de programmes (INF1035)
INF1020	Commerce électronique
INF1021	Les architectures réparties (INF1009)
INF1030	Introduction au développement d'applications mobiles (version Android) (INF1005 ou INF1035)
INF1032	Introduction au développement d'applications mobiles (plateforme IOS) (INF1005 ou INF1035)
INF1036	Sujets spéciaux en TI
PRO1027	Programmation scientifique en C (INF1002 ou PRO1026)
SDD1005	Probabilité & Inférence (SDD1002 ou STT1001 ou STT1042)
SIF1032	Graphisme par ordinateur (MAP1006 ou PIF1006; INF1004)
SIF1033	Traitement d'image (INF1004; STT1001 ou STT1042)

Cours complémentaires (9 crédits)
L'étudiant choisit neuf (9) crédits de cours complémentaires parmi les suivants. Certains cours sont offerts en ligne. Consulter la liste.

Autres renseignements

Règlements pédagogiques particuliers

Pour s'inscrire au cours SIF1015 Systèmes d'exploitation, les étudiants du baccalauréat en génie électrique (concentration génie informatique) (7144) doivent avoir réussi le cours GEI1072 Résolution de problèmes d'ingénierie en C et les étudiants du baccalauréat en informatique (7833) doivent avoir réussi les cours INF1004 Structures de données et algorithmes et SIF1053 Architecture des ordinateurs.

Pour s'inscrire au cours TIN1003 Science, technologie et société (2 crédits) et INF1014 Aspects juridiques de l'informatique (1 crédit), l'étudiant doit avoir complété 60 crédits dans le programme.

Pour s'inscrire au cours INF1022 Projet de synthèse en informatique, l'étudiant doit avoir réussi 60 crédits du programme.

Pour s'inscrire au cours INF1015 Stage d'informatique I, l'étudiant doit avoir réussi obligatoirement 45 crédits du programme.

Pour s'inscrire au cours INF1016 Stage d'informatique II, l'étudiant doit avoir réussi obligatoirement 60 crédits du programme et le cours INF1015 Stage d'informatique I.

Exceptionnellement, l'étudiant qui n'a pas eu la possibilité de s'inscrire aux cours INF1015 Stage d'informatique I et/ou INF1016 Stage d'informatique II devra compléter son programme en s'inscrivant au cours INF1022 Projet de synthèse en informatique et/ou à un ou deux cours optionnels du programme.

Équivalences et reconnaissance des acquis

Reconnaissance des acquis

L'étudiant admis au DEC-BAC en informatique qui aura complété la formation collégiale conformément à la liste des cours et aux autres exigences prévues aux ententes entre l'UQTR et les cégeps de Trois-Rivières, de Drummondville, de Shawinigan et de Sorel-Tracy se verra reconnaître, sous forme d'exemption, un total de trente (30) crédits.

N.B.: Les crédits reconnus par exemption apparaissent sur le relevé de note de l'étudiant avec la mention K. Ils n'entrent pas dans le calcul de la moyenne cumulative.

Description des activités

IAR1001 Intelligence artificielle (STT1001 ou STT1003; INF1008; INF1004)

L'objectif principal de ce cours est d'apprendre les concepts de base de l'intelligence artificielle (IA) et, en particulier, ses aspects informatiques et mathématiques. L'étudiant devra être en mesure d'identifier les problèmes qui relèvent de l'IA, de sélectionner les techniques ou approches pertinentes et de les mettre en application de façon efficiente. Les techniques de représentation des connaissances (structures de données) et de recherche (algorithmes), ainsi que leur réalisation informatique, de même que certains domaines d'application, constituent la matière fondamentale de ce cours. Plus spécifiquement, la matière sera présentée principalement dans la perspective des agents intelligents, et, de façon complémentaire, dans le contexte des systèmes multiagents et de l'intelligence artificielle distribuée.

Le cours fait le tour des aspects d'actualité de l'intelligence artificielle. Les agents "intelligents" servent de vecteur pour promouvoir l'explication des autres concepts de l'IA. Parmi ces concepts, nous voyons : les méthodes et stratégies de recherches de solution, les méthodes de jeux, la connaissance et le raisonnement fondés sur la logique du premier ordre et les inférences en logique du premier ordre, la planification et l'action logique et, enfin, le raisonnement probabiliste avec des connaissances incertaines. Dans une deuxième partie, nous introduisons la notion d'apprentissage automatique au moyen des réseaux de neurones et le traitement des langues naturelles.

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 (INF1002)

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 (INF1004)

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 (INF1006)

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.

INF1008 Analyse et conception d'algorithmes (ALG1001 ou PIF1005; INF1004)

L'étudiant développera son habileté à produire des algorithmes corrects et efficaces. Il pourra analyser la complexité spatiale et temporelle d'un algorithme. Il entrera en contact avec différentes classes d'algorithmes et s'habituera à choisir le type d'algorithme qui convient le mieux à un problème donné.

Définition d'un algorithme et de sa complexité. Notations asymptotiques. Résolution de récurrence. Étude d'efficacité et de complexité d'algorithmes de tri. Stratégies de conception d'algorithmes : algorithmes voraces, diviser-pour-régner, programmation dynamique, algorithmes probabilistes, exploration de graphes. Études de cas.

INF1009 Réseaux d'ordinateurs I (SIF1015)

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.

INF1010 Réseaux d'ordinateurs II (INF1009)

Ce cours permet aux étudiants de se familiariser avec les concepts et les contraintes des nouvelles technologies de l'information et des communications.

Présentation des spécificités du réseau Internet : services, protocoles, applications, administration; introduction à la gestion sécurisée de l'information et aux techniques et mécanismes de sécurité informatique.

Ce cours offre la possibilité aux étudiants de définir, de spécifier et de réaliser des applications clients serveurs sur le réseau Internet et de faire de l'administration de quelques services TCP/IP (DHCP, DNS,…) et des outils réseaux (Switch, routeur,..).

INF1011 Génie logiciel (INF1007)

Familiariser l'étudiant avec les concepts et principes fondamentaux du génie logiciel et le sensibiliser à leur importance dans la production de logiciels de qualité. Amener l'étudiant à une compréhension des principes de base de l'assurance qualité.

Principes et concepts fondamentaux du Génie Logiciel. Produit logiciel: caractéristiques. Processus de génie logiciel. Modèles de cycle de vie. Prototypage. Normes. Maturité des processus. Processus léger. Méthodes Agiles. Réutilisation de logiciel. Concepts favorisant la réutilisation - Pattern, COO, paradigme aspect. Entités réutilisable - lignes de produits logiciels, composants, produits COTS. Assurance qualité logiciel : Principes et Techniques. Processus de validation et de vérification : définition, rôles, techniques, impact sur la qualité, etc. Processus de test : techniques et stratégies. Planification, conception et réalisation des tests. Techniques de revues, Preuves, etc. Concepts et principes de la qualité. Contrôle de la qualité : qualité et métriques. Gestion de projets. Maintenance. Réingénierie. Documentation technique.

INF1012 Élaboration et gestion de projets scientifiques (INF1011)

Identifier et analyser les conséquences de l'innovation technologique et de la recherche et développement en sciences. Acquérir des connaissances et des techniques de travail en situation de projets informatiques. Développer l'aptitude à planifier de tels projets, à y participer et, éventuellement, à les gérer. Développer l'habileté à gérer son temps et celui de ses collaborateurs. Se familiariser à la dimension internationale de la gestion des projets.

Les différentes formes d'innovation technologique. Évaluation des impacts positifs et négatifs de l'innovation technologique et de la recherche et développement en sciences : étude coût/bénéfice, impact sur les individus, la communauté scientifique et la société. Gestion de projets informatiques : identification des objectifs et définition du projet, élaboration de solutions à caractère scientifique, planification des activités, échéanciers; choix de l'équipe, attribution des responsabilités; importance de la coordination en environnement de projet; suivi des activités et gestion du temps; progiciels de gestion de projets.

INF1013 Environnement de développement et composantes logicielles (INF1005 ou INF1035; SIF1015)

Compléter la formation des étudiants en lien avec les méthodologies et technologies de développement des applications Frame/Web d'envergure. Apprendre à construire des composants réutilisables en s'appuyant sur des bibliothèques existantes. Réaliser les applications Web et Frame selon les modèles MVC/MVP ainsi que des Application Serveur sous forme d'APE RESTFUL. Utiliser et

configurer des environnements de développement pour de grands projets logiciels. Maîtriser les gestionnaires de dépendances de projets. Open Source/Propriétaires. Quelques-unes des technologies importantes seront utilisées : C# - ASP.Net, Java-JSP, Python-DJANGO, Ruby-Rails, GWT, ANGULAR, REACT, Maven, Gradle, etc.

INF1014 Aspects juridiques de l'informatique (1 crédit)

L'étudiant se familiarise avec les aspects juridiques de l'informatique. Il apprend à connaître les principales lois touchant la pratique de l'informatique. Il développe l'habileté à analyser des situations de travail de façon à éviter les problèmes juridiques.

Notions juridiques : juridiction et territorialité, droit civil et droit pénal. Principales lois touchant la pratique de l'informatique : le droit d'auteur, le respect de la vie privée dans les secteurs public et privé, l'accès à l'information, le cadre juridique des technologies de l'information. Les aspects légaux des licences de produits informatiques. Les aspects légaux particuliers à l'Internet.

Règlements pédagogiques particuliers : Pour s'inscrire aux cours TIN1003 Science, technologie et société et INF1014 Aspects juridiques de l'informatique, l'étudiant en informatique doit avoir complété 60 crédits dans le programme.
Pour s'inscrire à ce cours, l'étudiant inscrit au double baccalauréat en mathématiques et en informatique ou au double baccalauréat en physique et en informatique doit avoir complété 90 crédits dans le programme.

INF1015 Stage d'informatique I

L'objectif du cours Stage d'informatique I est de permettre à l'étudiant d'appliquer ses connaissances en informatique en milieu de travail.

La durée normale du stage est de 12 à 16 semaines à temps plein.

Règlement pédagogique particulier : Pour s'inscrire au cours INF1015 Stage d'informatique I, l'étudiant doit avoir réussi obligatoirement 45 crédits du programme.

INF1016 Stage d'informatique II

Les objectifs du cours Stage d'informatique II sont de permettre à l'étudiant d'appliquer ses connaissances en informatique en milieu de travail et de parfaire l'acquisition des méthodes de travail propres à l'informatique.

La durée normale du stage est de 12 à 16 semaines à temps plein.

Règlement pédagogique particulier : Pour s'inscrire au cours INF1016 Stage d'informatique II, l'étudiant en informatique doit avoir réussi obligatoirement 60 crédits du programme et le cours INF1015 Stage d'informatique I.
Pour s'inscrire au cours INF1016 Stage d'informatique II, l'étudiant inscrit au double baccalauréat en mathématiques et en informatique ou au double baccalauréat en physique et en informatique doit avoir réussi obligatoirement 90 crédits du programme et le cours GMA1001 Stages.

INF1018 Analyse de programmes (INF1035)

Présenter à l'étudiant les principes et les techniques de base de l'implantation des langages de programmation. L'initier à l'évaluation des langages de programmation et à l'analyse de programmes. Familiariser l'étudiant avec certains outils de description et d'analyse des langages de programmation. Étude des langages de programmation dans l'optique de construction d'outils d'environnement de programmation.

Concepts fondamentaux des langages de programmation modernes. Classification et évaluation des langages de programmation. Concepts et critères de qualité. Syntaxe et sémantique des langages de programmation. Méthodes de description formelles. Grammaire et Langages. Méthodes d'analyse lexicale et syntaxique : principes, approches et applications. Outils pour le traitement des langages. Analyse sémantique. Introduction à la génération et à l'optimisation de code. Éléments d'analyse de programmes en langages évolués : statique, dynamique, dépendances, etc.

INF1020 Commerce électronique

Ce cours a comme objectif principal l'initiation aux concepts de base des affaires et du commerce électronique et leur mise en pratique dans la réalisation d'un projet de commerce électronique. Le cours a aussi comme objectifs secondaires : d'apprendre à gérer un projet d'équipe en informatique.

Aperçu du commerce électronique; la "digitalisation" des affaires; les processus et stratégies B2B (business-to-business); le rôle

des intermédiaires indépendants (third parties); l'environnement régulateur; EDI, e-commerce et Internet; risques des systèmes non-sécuritaires; risques de gestion; normes, protocoles et langages Internet; cryptographie et authentification; murs coupe-feu (firewalls); mécanismes de paiement en ligne pour le e-commerce; agents intelligents; le marketing sur le Web; autres sujets d'actualité en affaires et en commerce électronique.

Règlement pédagogique particulier : Pour s'inscrire au cours INF1020 Commerce électronique, les étudiants inscrits au cheminement DEC-BAC (7533) doivent avoir réussi les cours obligatoires de la première session. Les étudiants inscrits au double baccalauréat : mathématiques et informatique (6833) ou inscrits au double baccalauréat en physique et en informatique (6925) doivent avoir réussi les cours obligatoires des cinq (5) premières sessions.

INF1021 Les architectures réparties (INF1009)

Ce cours permet d'acquérir des connaissances sur les systèmes répartis. Il permet aux étudiants de s'initier aux méthodes de conception et de réalisation des applications informatiques réparties. À la fin du cours, les étudiants seront capables de comparer plusieurs méthodes de réalisation de systèmes répartis.

Utilisation des SOCKETS : rappel sur le protocole TCP; interface Java de manipulation des sockets (Java.net); réalisation d'applications clients serveurs. Utilisation de Java RMI : introduction à RMI (Remote Method Invocation); implémentation des objets, côté client et côté serveur. Introduction à CORBA : évolution vers les applications distribuées; protocole RPC (Remote Procedure Calls); norme CORBA; implémentation du client et de l'objet distribué CORBA. Agents intelligents : concept d'agent; agents coopératifs, rationnels, adaptatifs, mobiles; agents informationnels intelligents; agents pour le Web; sécurité, mobilité, aspects algorithmiques. XML (eXtensible Markup Language) : SGML, HTML, XML et le Web; EDI et XML pour les applications d'affaires électroniques; définitions de nouveaux langages basés sur XML; technologies et applications XML; protocoles de communication basés sur le XML.

INF1030 Introduction au développement d'applications mobiles (version Android) (INF1005 ou INF1035)

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.

INF1031 Développement d'applications mobiles avancées (version Android) (INF1030)

A partir de notions acquises dans le cours Introduction au développement d'applications mobiles (INF1030), ce cours permet aux étudiants de développer leurs compétences en développement d'applications mobiles avancées et ce dans des domaines divers comme les sciences, le commerce, la santé, le divertissement, etc. De plus, les étudiants seront amenés à mieux comprendre le développement d'applications verticales impliquant des notions d'architecture multi-tiers, de connectivité réseau. De sécurité, etc.

INF1032 Introduction au développement d'applications mobiles (plateforme IOS) (INF1005 ou INF1035)

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 (INF1002)

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 (INF1004)

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.

INF1036 Sujets spéciaux en TI
Développement de logiciel :

Ce cours a pour objectif d'ajouter au programme des activités revêtant un intérêt actuel en développement de logiciel et qui ne sont pas nécessairement vues ou approfondies dans le programme.

Développement d'applications mobiles :

Ce cours a pour objectif d'ajouter au programme des activités revêtant un intérêt actuel en développement d'applications mobiles et qui ne sont pas nécessairement vues ou approfondies dans le programme.

Science des données :

Ce cours a pour objectif d'ajouter au programme des activités revêtant un intérêt actuel en science des données et qui ne sont pas nécessairement vues ou approfondies dans le programme.

Cybersécurité et infonuagique :

Ce cours a pour objectif d'ajouter au programme des activités revêtant un intérêt actuel en cybersécurité et infonuagique et qui ne sont pas nécessairement vues ou approfondies dans le programme.

PIF1005 Mathématiques pour informaticiens I

Ce cours vise à introduire les concepts mathématiques fondamentaux en informatique.

Rappels mathématiques : calcul avec les nombres réels, inégalités, logarithmes; théorie des ensembles : relations, fonctions, relations d'ordre; logique : techniques de preuve; algèbre booléenne ; graphes; induction mathématique.

PIF1006 Mathématiques pour informaticiens II (PIF1005)

Ce cours vise à compléter la formation de base en mathématiques en vue d'applications en informatique.

Algèbre matricielle : opérations, déterminant et matrice inverse; concepts d'analyse numérique : représentation des nombres et erreur, interpolation et approximation; langages formels : langage et grammaire, machine à états finis avec et sans sorties , équivalence d'automates, automate fini non déterministe, expression régulière, machine de Turing; cryptographie : schéma de cryptage, authentification, identification, cryptographie à clé secrète, cryptographie à clé publique; codage et compression de données : techniques de compression, compression sans perte (code de Hoffman), compression avec perte (compression JPEG).

Ce cours comporte des travaux avec le logiciel Matlab.

PRO1027 Programmation scientifique en C (INF1002 ou PRO1026)

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 (STT1001; PIF1005)

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 (SMI1002; INF1004)

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.

SDD1004 Apprentissage Automatique et applications (SDD1002)

Cadre théorique et pratique des algorithmes d'apprentissage automatique (comme machine à vecteurs de support (SVM)) entraînement, validation. Introduction à la modélisation et aux algorithmes d'apprentissage machine basée sur les données (apprentissage par l'exemple). Algorithmes: voisin le plus proche (comme KNN), régression linéaire et linéaire généralisée, mixtures de Gaussiennes, processus Gaussien, méthodes de noyaux. Introduction à l'apprentissage profond.

SDD1005 Probabilité & Inférence (SDD1002 ou STT1001 ou STT1042)

Ce cours aborde les différentes techniques théoriques et pratiques de la décision, Files d'attente, chaînes de Markov et réseaux Bayésiens.

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.

SIF1032 Graphisme par ordinateur (MAP1006 ou PIF1006; INF1004)

L'étudiant se familiarise avec les algorithmes fondamentaux du graphisme : représentation des objets 2D et 3D. Il est amené à comprendre les concepts de transformation linéaire en coordonnées homogènes, la composition des transformations et le calcul matriciel, la transformation en perspective, les modèles de réflexion de la lumière, la couleur, l'illumination et le rendu, l'ajout de texture.

Création de modèles d'objets simples, formes polygonales; création de modèles plus complexes à l'aide de moteurs de formes; application de transformations 3D pour la visualisation et les déplacements d'objets 3D; introduction aux notions de réalisme d'une scène : illumination, couleur, propriétés des objets : texture, transparence.

Le cours comporte des projets pratiques d'animation réalisés en langage C, en langage C++ à l'aide des libraires graphiques OpenGl et DirectX

SIF1033 Traitement d'image (INF1004; STT1001 ou STT1042)

Amener l'étudiant à pouvoir utiliser l'ordinateur pour traiter une image de façon à en améliorer la perception visuelle et à permettre la détection d'objets significatifs.

Présentation des notions de base du traitement de l'image : correction géométrique et radiométrique des images, amélioration du

contraste, amélioration des images par filtrage spatial isotropique et anisotropique, spectral et morphologique; segmentation des images par seuillage; détection des arêtes et des contours, détection des droites et des formes circulaires, détection des objets par corrélation spatiale; génération d'images tomographiques; applications Client/Serveur permettant le traitement d'image à distance; protocoles de transfert d'images.

Les étudiants expérimentent les concepts du traitement des images par des projets pratiques codés en langage C ou en langage C++.

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.

SMI1002 Bases de données II (SMI1001)

Le cours porte principalement sur des considérations techniques sous-jacentes à l'utilisation ou à la conception des SGBD, toujours dans le contexte du SGBD Oracle qui sert d'illustration et de contexte de mise en pratique des concepts. L'étudiant aura donc l'opportunité d'approfondir sa compréhension du fonctionnement interne d'un SGBD, notamment de Oracle, et d'autres aspects complémentaires associés à l'exploitation des données.

Stockage et structures de fichiers, indexation et fonctions de hachage. Traitement et optimisation des requêtes. Traitement des transactions. Contrôle de la concurrence. Systèmes de recouvrement. Sécurité, contrôle des accès et encryptage. Bases de données et applications Web, modèles client-serveur et multitiers. Architectures des SGBD, bases de données parallèles et distribuées. Analyses des données, OLAP (on-line analytical processing), entreposage (warehousing) et forage (mining) de données.

Ce cours utilise le système de gestion de bases de données Oracle, le langage PL/SQL et l'interface JSP (Java Sever Page).

STT1001 Probabilités et statistiques

Statistiques de base en vue des applications.

Séries statistiques : histogramme et polygone. Mesures de tendance centrale. Mesures de dispersion. Moments. Eléments de probabilités : variables aléatoires, distributions binomiales, hypergéométriques, normales. Poisson. Introduction à l'échantillonnage. Tests d'hypothèses simples.

TIN1003 Science, technologie et société (2 crédits)

Identifier les effets inducteurs et destructeurs des innovations scientifiques et technologiques sur la société contemporaine; provoquer une réflexion critique sur la viabilité sociale des technologies nouvelles; préparer les étudiants à accepter la responsabilité sociale qui incombe à la future « intelligence technique supérieure ».

Introduire aux thèmes principaux de la philosophie de la technique en étudiant les visages multiples de la technologie moderne et en examinant ses rapports avec les différentes dimensions impliquées dans le concept « qualité de vie ». Présenter une méthodologie pour l'analyse de risque : formes de risque (physiques, personnels, sociaux), méthodes d'évaluation de choix

technologiques et de détermination de la viabilité sociale de ces technologies. Proposer et discuter des modèles de prises de décision dans l'allocation de ressources technologiques et l'implantation de technologies nouvelles (avec applications particulières à l'informatique, la robotique, etc.). Identifier les principes fondamentaux devant être respectés dans l'implantation de ces technologies. Circonscrire la fonction nouvelle et la responsabilité sociales de l'ingénieur.

Règlement pédagogique particulier : Pour s'inscrire à ce cours, l'étudiant en informatique doit avoir réussi 60 crédits du programme.
Pour s'inscrire à ce cours, l'étudiant inscrit au double baccalauréat en mathématiques et en informatique ou au double baccalauréat en physique et en informatique doit avoir complété 90 crédits dans le programme.