

**Grade: Bachelier ès sciences (B.Sc.)****Crédits: 123**

## Présentation

### En bref

#### Objectifs du programme

##### Objectifs

Le programme de double baccalauréat en mathématiques et en informatique de l'Université du Québec à Trois-Rivières poursuit différents objectifs.

Au niveau informatique, l'orientation appliquée de la formation veut :

- Permettre à l'étudiant d'acquérir des connaissances en informatique fondamentale axées principalement sur la conception et la production de logiciels. Ainsi, le diplômé aura une solide formation de base fondée sur l'application des notions fondamentales des sciences informatiques pour le développement et la gestion de logiciels de qualité, à partir de l'analyse des besoins jusqu'à leur maintenance. Le diplômé sera apte à travailler dans des domaines aussi variés que la réseautique, le traitement d'images, le multimédia, le développement d'applications Web, les bases de données, l'intelligence artificielle, le commerce électronique, etc.
- Développer la capacité d'intégration des connaissances acquises pour la résolution de problèmes dans les différents domaines cités. Par ailleurs, les stages en entreprise permettent à l'étudiant d'acquérir une expérience pratique de participation au développement d'applications dans les entreprises et favorisent l'intégration harmonieuse du diplômé au marché du travail.

Au niveau mathématique et statistique, le programme vise à:

- Donner à l'étudiant une solide formation de base en mathématiques par l'acquisition des connaissances et le développement des habiletés propres au mathématicien.
- Permettre à l'étudiant d'acquérir des compétences particulières en statistiques.
- Développer chez l'étudiant son intuition, sa créativité, son imagination et son esprit logique.
- La proximité des outils informatiques au département de mathématiques et d'informatique favorise l'apprentissage des logiciels mathématiques et des langages de programmation nécessaires aux applications du marché du travail.

#### Concentrations, profils, cheminements

Il est à noter que le programme en informatique propose deux cheminements permettant à l'étudiant de pouvoir se spécialiser soit dans le développement logiciel (01) ou soit le développement d'applications Web et mobiles (02).

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

- Cheminement 01 : Double baccalauréat : mathématiques et informatique (Développement de logiciel);
- Cheminement 02 : Double baccalauréat : mathématiques et informatique (Développement d'applications Web et mobiles).

Exclusif au Québec, ce cheminement d'une durée normale de quatre années conduit, par articulation des exigences propres aux deux professions, à l'obtention de deux grades de bachelier ès sciences (B.Sc. en mathématiques et B.Sc. en informatique).

---

Atouts UQTR

Support informatique

En plus de pouvoir utiliser l'infrastructure réseautique fournie par l'Université, les étudiants du programme d'informatique ont à leur disposition un réseau de micro-ordinateurs performants comportant les langages et les logiciels utilisés dans les cours. Ils ont aussi accès à des ordinateurs réseau puissants à partir du campus ou de leur domicile.

Les étudiants d'informatique ont l'occasion de travailler avec les outils modernes de programmation : les langages de programmation C, C++, C#, Java, les systèmes d'exploitation Windows, UNIX et Linux, Android et IOS ainsi qu'une variété d'environnements de développement intégrés (IDE) et d'appareils mobiles.

## **La recherche dans le domaine**

Laboratoires

Le Département de mathématiques et d'informatique dispose de quatre laboratoires pour les besoins de l'enseignement de l'informatique. Deux des laboratoires sont à vocation générale. Les étudiants peuvent y faire leurs travaux et avoir accès aux serveurs du département. Les professeurs peuvent utiliser un de ces laboratoires pour faire des démonstrations pendant les cours.

Un laboratoire est réservé au commerce électronique et à l'installation par les étudiants de logiciels serveurs.

Un dernier laboratoire permet aux étudiants d'expérimenter des sujets avancés sur la programmation réseau, sur les systèmes d'exploitation et sur plusieurs domaines d'applications dont le développement d'applications Web et mobiles.

Pendant les sessions d'enseignement, les laboratoires sont accessibles vingt-quatre heures par jour.

Logiciels

Les étudiants d'informatique et de mathématiques ont accès aux outils modernes de programmation, en l'occurrence : les langages de programmation C, C++, Java, les systèmes d'exploitation Windows, UNIX et Linux, ainsi qu'à une variété d'environnements de développement intégrés (IDE). Les étudiants peuvent installer plusieurs de ces produits dans leur propre environnement de travail.

Stages

Dans le cadre de son programme de formation, l'étudiant a l'occasion de mettre en pratique ses acquis en réalisant deux stages rémunérés en entreprise.

## **Admission**

**Trimestre d'admission et rythme des études**

Automne, hiver, été.

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

## **Conditions d'admission**

**Études au Québec**

Base DEC

Etre titulaire d'un diplôme d'études collégiales (DEC) ou l'équivalent et avoir complété les cours de niveau collégial suivants ou leur équivalent :

---

Mathématiques : 00UN ou 01Y1 ou 022X; 00UP ou 01Y2 ou 022Y; 00UQ ou 01Y4 ou 022Z.

OU

Etre titulaire d'un diplôme d'études collégiales (DEC) en « Sciences, lettres et arts » ou son équivalent.

Base universitaire

Etre titulaire d'un diplôme universitaire (baccalauréat ou certificat) ou l'équivalent et posséder des connaissances équivalentes au contenu des cours de niveau collégial suivants :

Mathématiques : 00UN ou 01Y1 ou 022X; 00UP ou 01Y2 ou 022Y; 00UQ ou 01Y4 ou 022Z.

Le candidat admissible dont on n'aura pu établir à l'aide du dossier qu'il possède toutes les connaissances requises pourra être admis conditionnellement à la réussite de cours d'appoint en mathématiques.

Base expérience

Etre âgé d'au moins vingt et un ans, avoir oeuvré pendant au moins un an dans un domaine relié aux mathématiques et posséder des connaissances équivalentes au contenu des cours de niveau collégial suivants :

Mathématiques : 00UN ou 01Y1 ou 022X; 00UP ou 01Y2 ou 022Y; 00UQ ou 01Y4 ou 022Z.

### **Etudes hors Québec**

Base études hors Québec

Etre détenteur d'un diplôme d'études préuniversitaires totalisant 13 années;

OU

d'un diplôme d'études préuniversitaires totalisant 12 années et une année d'études universitaires (à moins d'ententes conclues avec le Gouvernement du Québec, tous les candidats, ayant 12 ans de scolarité devront compléter une année de mise à niveau);

OU

d'un baccalauréat de l'enseignement secondaire français (général ou technologique).

ET

Posséder les équivalences suivantes Mathématiques : 00UN ou 01Y1 ou 022X; 00UP ou 01Y2 ou 022Y; 00UQ ou 01Y4 ou 022Z.

### **Conditions supplémentaires hors Québec**

Pour mener à bien vos études, une bonne maîtrise de la langue française est nécessaire. Ainsi, vous devez nous faire parvenir la copie originale de votre test de français reconnu par l'UQTR (TFI, TEF, TCF, DELF, DALF) ou vous inscrire à la passation du TFI de l'UQTR à l'adresse [www.uqtr.ca/inscription\\_tfr](http://www.uqtr.ca/inscription_tfr) Pour obtenir de l'information sur les tests de français reconnus et les seuils de réussite, veuillez consulter le lien suivant : <https://oraprdnt.uqtr.quebec.ca/vrsg/Reglementation/107.pdf>

## **Structure du programme et liste des cours**

### **Double bacc. maths-info (dével. logiciel)**

#### **(Cheminement: 1)**

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

#### **Cours obligatoires (99 crédits)**

ALG1001 Logiques et Ensembles

GMA1001	Stages
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)
INF1011	Génie logiciel (INF1007)
INF1014	Aspects juridiques de l'informatique (1 crédit)
INF1016	Stage d'informatique II
INF1034	Introduction aux interfaces utilisateur (INF1002)
INF1035	Concepts avancés en objet (INF1004)
MAP1003	Analyse numérique (INF1002 ou PRO1026; MAP1007)
MAP1006	Mathématiques appliquées I
MAP1007	Mathématiques appliquées II
MPU1015	Algèbre (ALG1001)
MPU1017	Algèbre linéaire et applications
MPU1018	Algèbre linéaire avancée (MPU1015)
MPU1027	Equations différentielles (MPU1045; MAP1006)
MPU1039	Analyse complexe (MPU1054; MPU1055)
MPU1045	Analyse à une variable réelle I
MPU1054	Analyse à une variable réelle II (MPU1045)
MPU1055	Topologie et analyse à plusieurs variables réelles (MPU1045)
ROP1021	Optimisation
SIF1015	Systèmes d'exploitation
SIF1053	Architecture des ordinateurs
SMI1001	Bases de données I
SMI1002	Bases de données II (SMI1001)
STT1003	Probabilités
STT1042	Statistique (STT1003)
TIN1003	Science, technologie et société (2 crédits)

### Cours optionnels (18 crédits)

#### L'étudiant doit suivre trois crédits parmi les cours suivants :

ALG1003	Applications algébriques (MPU1015)
GEM1001	Géométries euclidienne et non euclidienne (MPU1015)
MPU1056	Nombres et structures (ALG1001)
MPU1058	Mesure et intégration (MPU1054)
PMA1002	Sujets spéciaux en mathématiques I (MPU1018 ou MPU1055 ou STT1030)
PMA1003	Sujets spéciaux en mathématiques II (MPU1018 ou MPU1055 ou STT1030)
STT1020	Analyse de la variance
STT1028	Séries chronologiques (STT1030)
STT1030	Modèles de régression et de prévision (STT1042)
STT1032	Théorie et pratique des sondages (PAF1010 ou STT1001 ou STT1042)
STT1047	Statistique mathématique (STT1042)

#### L'étudiant doit suivre les neuf crédits de cours suivants :

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 crédits parmi les cours suivants :

IAR1001	Intelligence artificielle (STT1001 ou STT1003; INF1008; INF1004)
INF1033	Sujets spéciaux en développement de logiciel ou en développement d'applic. mobiles
PRO1027	Programmation scientifique en C (INF1002 ou PRO1026)
SIF1033	Traitement d'image (INF1004; STT1001 ou STT1042)

---

Cours complémentaires (6 crédits)

L'étudiant choisit deux cours (six crédits) parmi la liste de cours complémentaires suggérée. Certains cours sont offerts en ligne. Consulter la liste.

### **Double bacc. maths-info (dév. app. Web et mobiles)**

#### **(Cheminement: 2)**

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

#### **Cours obligatoires (99 crédits)**

ALG1001	Logiques et Ensembles
GMA1001	Stages
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)
INF1011	Génie logiciel (INF1007)
INF1014	Aspects juridiques de l'informatique (1 crédit)
INF1016	Stage d'informatique II
INF1034	Introduction aux interfaces utilisateur (INF1002)
INF1035	Concepts avancés en objet (INF1004)
MAP1003	Analyse numérique (INF1002 ou PRO1026; MAP1007)
MAP1006	Mathématiques appliquées I
MAP1007	Mathématiques appliquées II
MPU1015	Algèbre (ALG1001)
MPU1017	Algèbre linéaire et applications
MPU1018	Algèbre linéaire avancée (MPU1015)
MPU1027	Equations différentielles (MPU1045; MAP1006)
MPU1039	Analyse complexe (MPU1054; MPU1055)
MPU1045	Analyse à une variable réelle I
MPU1054	Analyse à une variable réelle II (MPU1045)
MPU1055	Topologie et analyse à plusieurs variables réelles (MPU1045)
ROP1021	Optimisation
SIF1015	Systèmes d'exploitation
SIF1053	Architecture des ordinateurs
SMI1001	Bases de données I
SMI1002	Bases de données II (SMI1001)
STT1003	Probabilités
STT1042	Statistique (STT1003)
TIN1003	Science, technologie et société (2 crédits)

#### **Cours optionnels (18 crédits)**

##### **L'étudiant doit suivre trois crédits parmi les cours suivants :**

ALG1003	Applications algébriques (MPU1015)
GEM1001	Géométries euclidienne et non euclidienne (MPU1015)
MPU1056	Nombres et structures (ALG1001)
MPU1058	Mesure et intégration (MPU1054)
PMA1002	Sujets spéciaux en mathématiques I (MPU1018 ou MPU1055 ou STT1030)
PMA1003	Sujets spéciaux en mathématiques II (MPU1018 ou MPU1055 ou STT1030)
STT1020	Analyse de la variance
STT1028	Séries chronologiques (STT1030)
STT1030	Modèles de régression et de prévision (STT1042)
STT1032	Théorie et pratique des sondages (PAF1010 ou STT1001 ou STT1042)
STT1047	Statistique mathématique (STT1042)

---

L'étudiant doit suivre les neuf crédits de 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)

**L'étudiant doit suivre six crédits parmi les cours suivants :**

- INF1021 Les architectures réparties (INF1009)
- INF1032 Introduction au développement d'applications mobiles (plateforme IOS) (INF1005 ou INF1035)
- INF1033 Sujets spéciaux en développement de logiciel ou en développement d'applic. mobiles
- SIF1032 Graphisme par ordinateur (MAP1006 ou PIF1006; INF1004)

### **Cours complémentaires (6 crédits)**

L'étudiant choisit deux cours (six crédits) parmi la liste de cours complémentaires suggérée. Certains cours sont offerts en ligne. Consulter la liste.

## **Autres renseignements**

### **Règlements pédagogiques particuliers**

Pour s'inscrire au cours INF1008 Analyse et conception d'algorithmes, l'étudiant doit avoir complété les cours obligatoires des 3 premiers trimestres du programme.

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 90 crédits dans le programme.

Pour s'inscrire au cours INF1022 Projet de synthèse en informatique, l'étudiant doit avoir réussi 90 crédits du programme.

Pour s'inscrire au cours GMA1001 Stages, l'étudiant doit avoir réussi obligatoirement 75 crédits du programme.

Pour s'inscrire au cours INF1016 Stage d'informatique II, l'étudiant doit avoir réussi obligatoirement 90 crédits du programme et le cours GMA1001 Stages.

Exceptionnellement, l'étudiant qui n'a pas eu la possibilité de s'inscrire au cours GMA1001 Stages devra compléter son programme en s'inscrivant au cours MPU1040 Projet de synthèse en mathématique et/ou celui qui n'a pas eu la possibilité de s'inscrire au cours INF1016 Stage d'informatique II devra compléter son programme en s'inscrivant au cours INF1022 Projet de synthèse en informatique et/ou à un cours optionnel en informatique du programme.

### **Description des activités**

#### **ALG1001 Logiques et Ensembles**

S'initier aux concepts de base des mathématiques actuelles à travers l'étude de concepts de la logique et de la théorie des ensembles. Développer la capacité de faire des preuves.

Logique propositionnelle : formes propositionnelles et connecteurs logiques, dérivations et méthodes de preuves. Logique des prédicats : quantificateurs et raisonnements avec ceux-ci. Concepts ensemblistes : approche intuitive et approche axiomatique des ensembles, axiome des naturels et preuve par induction mathématique. Relations, relations d'ordre, relations d'équivalence. Fonctions et applications.

#### **ALG1003 Applications algébriques (MPU1015)**

A partir de problèmes, développer et appliquer des concepts et des outils mathématiques pour modéliser et résoudre ces problèmes. Utiliser les grandes techniques mathématiques : l'abstraction, la généralisation, l'induction mathématique, les structures et ses isomorphismes, les ensembles-quotient pour solutionner des problèmes concrets. Explorer les liens entre différents domaines mathématiques tels l'algèbre et la logique, l'analyse, l'informatique...

Réseaux logiques et algèbre de Boole. Automates déterministes, réduction d'automates et automates non-déterministes. Problèmes de codage. Problèmes et thèmes choisis en théorie des nombres, des ensembles flous, des corps, en combinatoire...

---

## **GEM1001 Géométries euclidienne et non euclidienne (MPU1015)**

S'initier à l'approche moderne des mathématiques par l'étude axiomatique de géométries.

Systèmes axiomatiques de géométries. Géométries finies. Géométrie neutre (ou absolue), géométrie euclidienne et géométrie hyperbolique, initiation à la géométrie elliptique. Les automorphismes de la géométrie : les isométries et les homothéties.

### **GMA1001 Stages**

Acquérir une expérience pratique du travail de statisticien, de professeur ou d'informaticien dans une entreprise utilisant les statistiques et/ou l'informatique, ou dans une maison d'enseignement. Faciliter l'intégration dans le milieu de travail.

L'étudiant présente un projet soumis à approbation. Ce projet devra préciser l'équipe, le groupe ou l'entreprise dont les activités principales correspondent aux statistiques et/ou à l'informatique et/ou à l'enseignement, et ceci pour une durée minimale de 135 heures d'activités d'ordre technique. A la fin de son stage, l'étudiant doit soumettre un rapport de stage. Le stage sera sous la supervision d'un professeur du Département de mathématiques et d'informatique.

Règlement pédagogique particulier :

Pour les étudiants inscrits au baccalauréat en mathématiques (7721) : Avoir complété au moins 60 crédits du programme et avoir au moins 2,5 de moyenne cumulative.

Pour les étudiants inscrits au double bacc : mathématiques et enseignement au secondaire - mathématiques (6721) : Avoir complété au moins 75 crédits du programme et avoir au moins 2,5 de moyenne cumulative.

Pour les étudiants inscrits au double bacc : mathématiques et informatique (6833) : Avoir complété au moins 75 crédits du programme.

## **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 multi-agents 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'habitue 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.

### **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 meilleures 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 meilleures 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).

### **INF1033 Sujets spéciaux en développement de logiciel ou en développement d'applic. mobiles**

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.

### **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 pointcut, 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.

### **MAP1003 Analyse numérique (INF1002 ou PRO1026; MAP1007)**

Familiariser l'étudiant avec les méthodes numériques, les éléments d'analyse réelle et matricielle à la base des méthodes numériques et leur mise en oeuvre informatique.

Arithmétique computationnelle. Résolution des équations non linéaires. Résolution des systèmes linéaires et non linéaires de grande taille. Recherche des valeurs et vecteurs propres. Interpolation et lissage. Intégration et dérivation numérique. Éléments de la théorie de l'approximation et du traitement du signal (transformées). Méthodes numériques en optimisation.

### **MAP1006 Mathématiques appliquées I**

Transmettre à l'étudiant les bases de l'algèbre matricielle, l'introduire aux méthodes numériques, lui donner les outils nécessaires à la résolution des équations différentielles ordinaires et lui montrer certaines applications des équations différentielles. Algèbre matricielle : matrices, définitions et opérations, matrice triangulaire, diagonale, transposée d'une matrice, matrice régulière et rang, déterminants, inverse d'une matrice, solution d'équations linéaires, valeurs et vecteurs propres. Nombres complexes. Équations différentielles : classification, solution d'une équation différentielle avec interprétation géométrique; équations différentielles du premier ordre, équations exactes et facteur intégrant, équations à variables séparables, homogènes, linéaires, de Bernoulli; applications (trajectoires orthogonales, problèmes de taux, etc.). Équations différentielles d'ordre supérieur : système fondamental de solutions, équations linéaires, homogènes à coefficients constants, réduction d'ordre, équations linéaires non-homogènes, équations d'Euler-Cauchy; résolution en séries de puissances; applications. Systèmes d'équations différentielles homogènes et non homogènes.

### **MAP1007 Mathématiques appliquées II**

Acquisition de notions fondamentales sur les fonctions à plusieurs variables, les courbes et les surfaces dans le plan et l'espace. Applications du calcul différentiel et intégral à plusieurs variables ainsi que le calcul vectoriel.

Éléments de géométrie analytique. Courbes planes, tangentes et longueur d'arc, coordonnées polaires, intégrales en coordonnées polaires, équations polaires des coniques. Vecteurs de dimension deux et trois, produit scalaire, produit vectoriel, droites et plans, surfaces. Fonctions vectorielles et courbes dans l'espace, limites, dérivées et intégrales, cinématique, courbure, composantes tangentielle et normale de l'accélération, lois de Kepler. Fonctions de plusieurs variables, limites et continuité, dérivées partielles, accroissement et différentielle, dérivation de fonctions composées, dérivées directionnelles, normales et plans tangents, valeurs extrêmes de fonctions à plusieurs variables, multiplicateurs de Lagrange. Intégrales doubles, aire et volume, intégrales doubles en coordonnées polaires, aires de surfaces, intégrales triples, moment d'inertie et centre de masse, coordonnées cylindriques, coordonnées sphériques, changement de variables et jacobiens. Champs vectoriels, intégrales curvilignes, indépendance du chemin, théorème de Green-Riemann, intégrales de surface, théorème de flux-divergence, théorème de Stokes.

### **MPU1015 Algèbre (ALG1001)**

Développer l'habileté à généraliser à travers l'étude des structures algébriques. Développer la capacité de démontrer et d'appliquer dans un contexte abstrait. Connaître les concepts et les méthodes de base de l'algèbre moderne. Comprendre les liens entre l'algèbre classique (du secondaire) et l'algèbre moderne.

Les structures de demi-groupe et de monoïde. La structure de groupe et les concepts connexes : sous-groupe, groupe cyclique, groupe-produit, groupe-quotient, homomorphisme de groupes et les théorèmes fondamentaux. La structure d'anneau et les concepts connexes : sous-anneau, anneau-produit, anneau-quotient, domaine d'intégrité, corps, anneau des polynômes sur un corps, corps des complexes.

### **MPU1017 Algèbre linéaire et applications**

Acquérir la connaissance des principaux concepts, problèmes, méthodes et applications de la théorie des espaces vectoriels et des transformations linéaires. Utilisation d'un logiciel de calcul symbolique.

Matrices et transformations, déterminants, espaces et sous-espaces de  $R^n$ , orthogonalité, systèmes d'équations linéaires, espaces vectoriels, applications linéaires, représentations matricielles, changement de base.

### **MPU1018 Algèbre linéaire avancée (MPU1015)**

Approfondir la théorie des espaces vectoriels et des transformations linéaires en situant les problèmes dans un cadre plus général et en les analysant plus en détail.

---

Valeurs propres et vecteurs propres d'un opérateur linéaire, diagonalisation, théorème de Cayley-Hamilton, formes canoniques, formes linéaires, espace dual, formes bilinéaires, quadratiques et hermitiennes, espaces euclidiens, opérateurs linéaires sur un espace euclidien.

**MPU1027 Equations différentielles (MPU1045; MAP1006)**

Méthodes analytiques dans l'obtention des solutions et dans l'étude qualitative des équations différentielles ordinaires.

Equations différentielles du premier ordre. Equations linéaires d'ordre quelconque à coefficients constants. Solutions en série. Problème de Sturm-Liouville et fonctions orthogonales.

**MPU1039 Analyse complexe (MPU1054; MPU1055)**

Développer la compréhension des éléments de la théorie des variables complexes et l'habileté à l'appliquer.

Les nombres complexes, dérivation complexe, équations de Cauchy-Riemann, intégration complexe (théorème de Cauchy), formule intégrale de Cauchy et ses conséquences, séries de Taylor et de Laurent, étude des points singuliers, calcul des résidus, calcul d'intégrales réelles et de séries réelles à l'aide des nombres complexes.

**MPU1045 Analyse à une variable réelle I**

Etudier les propriétés des nombres réels et de la topologie de  $\mathbb{R}$ . Développer la compréhension de la théorie du calcul différentiel et l'habileté à faire des démonstrations rigoureuses.

Le système des nombres réels, quelques concepts topologiques (théorème de Bolzano-Weierstrass), suites numériques, séries numériques, limite et continuité (théorème des valeurs intermédiaires) dérivation (théorème de Rolle, de la moyenne, règle de l'Hôpital).

**MPU1054 Analyse à une variable réelle II (MPU1045)**

Etudier la théorie de l'intégrale Riemann. Approfondir la compréhension du calcul intégral et des séries de fonctions.

Continuité uniforme, intégrale de Riemann (théorème fondamental du calcul, intégrales impropres), suites de fonction, séries de fonction, séries de puissances (séries de Taylor).

**MPU1055 Topologie et analyse à plusieurs variables réelles (MPU1045)**

Approfondir certaines notions de l'analyse réelle par la généralisation du concept de distance et l'étude des fonctions de plusieurs variables réelles. Explorer les notions de base de la topologie via les espaces métriques. Développer l'habileté à penser dans un contexte multidimensionnel.

Espaces métriques, espaces complets (théorème de point fixe de Banach), espaces connexes, espaces compacts, structure topologique, dérivées en plusieurs variables réelles, théorème de Taylor, problèmes d'extremum, fonctions inverses.

**MPU1056 Nombres et structures (ALG1001)**

Explorer la notion de nombre et les propriétés des systèmes de nombres. Découvrir la notion de structure algébrique et identifier les structures des ensembles des nombres.

Construction algébrique des naturels, des entiers et des rationnels. Propriétés de l'addition et de la multiplication des naturels, des entiers. Relation de la division entière, algorithme d'Euclide, notion de nombre premier et théorème fondamental de l'arithmétique, pgcd et ppcm. Relation de congruence, ses propriétés, construction des ensembles  $\mathbb{Z}/n\mathbb{Z}$  et de leurs propriétés, diviseurs de zéro. Notion de l'infini : ensembles équipotents, cardinaux. Étude des nombres complexes.

**MPU1058 Mesure et intégration (MPU1054)**

Approfondir les connaissances en analyse. Etudier une nouvelle approche d'intégration basée sur l'intégrale de Lebesgue.

Notions de mesures et tribus. Fonctions mesurables. Intégrale de Lebesgue et ses propriétés. Convergence monotone et dominée. Comparaison avec l'intégrale de Riemann.

**PMA1002 Sujets spéciaux en mathématiques I (MPU1018 ou MPU1055 ou STT1030)**

Ce cours a pour objectif d'ajouter au programme des activités revêtant un intérêt actuel en mathématique.

---

Les mathématiques continuent toujours d'innover et il est important que les étudiants aspirant à une carrière en mathématique aient l'occasion de connaître d'autres sujets que ceux vus actuellement dans le programme. Le département de mathématiques et informatique dispose d'une équipe de professeurs pouvant aborder de nombreux sujets intéressants comme : logique, combinatoire, analyse fonctionnelle, analyse en plusieurs variables réelles, fractales, théorie du chaos, histoire moderne de mathématiques, théorie des groupes, théorie des catégories, équations différentielles, théorie des risques et ses applications (actuariat, finance), etc.

### **PMA1003 Sujets spéciaux en mathématiques II (MPU1018 ou MPU1055 ou STT1030)**

Ce cours a pour objectif d'ajouter au programme des activités revêtant un intérêt actuel en mathématique.

Les mathématiques continuent toujours d'innover et il est important que les étudiants aspirant à une carrière en mathématique aient l'occasion de connaître d'autres sujets que ceux vus actuellement dans le programme. Le département de mathématiques et informatique dispose d'une équipe de professeurs pouvant aborder de nombreux sujets intéressants comme: logique, combinatoire, analyse fonctionnelle, analyse en plusieurs variables réelles, fractales, théorie du chaos, histoire moderne de mathématiques, théorie des groupes, théorie des catégories, équations différentielles, théorie des risques et ses applications (actuariat, finance), etc.

### **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.

Etude 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.

### **ROP1021 Optimisation**

Connaître les problèmes et les méthodes de la programmation linéaire et de la théorie des graphes et des réseaux. Devenir habile à modéliser. Comprendre le fonctionnement des algorithmes utilisés et être capable de les appliquer.

Programmation linéaire : les problèmes de programmation linéaire et leur modélisation, résolution graphique et matricielle, algorithmes du simplexe, notions de dualité et analyse de sensibilité. Graphes et réseaux : généralités sur les graphes, algorithmes de chemins, problèmes d'ordonnancement, flots de valeur maximale et coût minimum, problèmes d'affectation, autres applications.

### **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 cryptage. Bases de données et applications Web, modèles client-serveur et multitières. 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).

**STT1003 Probabilités**

Familiariser l'étudiant avec les notions de base de la théorie des probabilités.

---

Lois de probabilités; probabilités conditionnelles et indépendance; théorème de Bayes. Variables aléatoires et espérance mathématique. Lois de probabilités discrètes et continues : binomiale, Poisson, géométrique, hypergéométriques, uniforme, exponentielle, normale. Transformation de variables aléatoires. Probabilités et fonctions de densité jointes, marginales et conditionnelles. Approximation d'une loi binomiale : par une loi de Poisson, par une loi normale. Inégalité de Tchebycheff. Théorème limite central et applications.

**STT1020 Analyse de la variance**

Initier l'étudiant aux notions de l'analyse de la variance et de la planification d'expériences en vue d'applications.

Notions d'erreur expérimentale, de randomisation. Répliques. Blocs. Effets fixes et aléatoires. Modèle à un facteur. Comparaisons multiples. Modèle à deux facteurs. Interaction. Études de quelques plans d'expériences : plan complètement aléatoire, blocs aléatoires, carrés latins, incomplets. Résidus et diagnostics. Applications.

Afin de bien réussir ce cours, il est recommandé aux étudiants d'avoir suivi au moins un cours de statistique.

**STT1028 Séries chronologiques (STT1030)**

Étudier les bases théoriques des séries chronologiques. Développer l'aptitude à utiliser ces connaissances dans des cas concrets et la capacité de choisir le modèle qui convient à une situation donnée.

Intérêts et applications des séries chronologiques. Revue des principales méthodes : lissage et décomposition, régression simple et multiple, modèles autorégressifs à moyennes mobiles. Comparaison de ces méthodes. Aspects pratiques de la prédiction. Applications à des cas concrets et usage de programmes d'ordinateur.

**STT1030 Modèles de régression et de prévision (STT1042)**

Permettre à l'étudiant d'effectuer la conception et l'analyse de modèles statistiques linéaires à l'aide des techniques de la régression. Familiariser l'étudiant à l'emploi de certains programmes statistiques de la banque SPSS, BMD ou SAS. La régression sera considérée comme outil de prévision. A cela, s'ajoutera l'étude des méthodes prévisionnelles.

Estimation. Régression linéaire simple. Propriétés et distributions des estimateurs  $b_{0/SUB}$  et  $b_{1/SUB}$ . Prévision. Extrapolation. Test de linéarité. Normalité des résidus. Variance constante. Corrélation linéaire, transformation de Fisher. Modèles linéaires après transformation. Régression multiple : introduction, domaines d'application, modèles linéaires à plusieurs variables explicatives. Matrice des variances-covariances, corrélation, distribution des coefficients de régression, estimation ponctuelle et par intervalle. Colinéarité et différents tests d'hypothèses. Choix du meilleur ensemble de variables explicatives. Utilisation des programmes BMD, SPSS ou SAS. Utilisation de l'information qualitative : variables auxiliaires. Modèles de régression polynomiale. Influence interactive dans un modèle de régression. Modèles de régression plus élaborés. Expériences factorielles. Modèles prévisionnels : composantes d'une série chronologique, techniques prévisionnelles, modèle autorégressif, lissage exponentiel, modèles avec tendance et influence.

**STT1032 Théorie et pratique des sondages (PAF1010 ou STT1001 ou STT1042)**

Familiariser et initier l'étudiant aux principales méthodes de sondage, le sensibiliser aux problèmes relatifs à l'organisation d'un sondage, à la construction de questionnaires et à la non-réponse.

Rappel de notions probabilistes. Échantillon aléatoire simple sans remise. Échantillon stratifié. Estimateur-quotient. Sondage en grappes. Sondage à deux degrés. Problèmes relatifs à l'organisation d'un sondage. Contacter les gens, problèmes reliés aux enquêteurs. Problème de non-réponse et représentativité. Traitement informatique.

**STT1042 Statistique (STT1003)**

Familiariser l'étudiant avec les fondements et les concepts de l'inférence statistique générale. Initier l'étudiant aux logiciels statistiques.

Statistiques descriptives : représentations graphiques, mesures de tendance centrale et de dispersion. Distribution des fonctions et transformations de variables aléatoires. Distributions échantillonales : loi normale, loi de Student, loi khi-deux, Loi de Fisher. Méthodes d'estimation classique. Principales qualités des estimateurs. Estimation ponctuelle et par intervalles de confiance. Tests d'hypothèses : tests d'adéquation, tests d'hypothèses pour une moyenne, une proportion, une variance, deux moyennes, deux proportions et deux variances. Erreurs de première et de deuxième espèce; puissance d'un test.

**STT1047 Statistique mathématique (STT1042)**

Ce cours vise à initier l'étudiant aux différentes techniques de mathématique statistique.

Théorèmes limites. Distributions d'échantillonnage. Estimation paramétrique. Notions de statistique suffisante et de l'efficacité. Lemme de Neyman-Pearson, tests basés sur la fonction de vraisemblance. Méthodes paramétriques et non-paramétriques pour la comparaison de deux échantillons. Notions de base pour le plan d'expérience, l'analyse de tableaux de contingence, la théorie de

---

la décision et l'inférence bayésienne.

**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.