Catalogue des cours sélectionnés

1 : INF311 Introduction à l'informatique
2 : INF431 Algorithmique et programmation
3 : INF421 Les bases de la programmation et de l'algorithmique
4 : INF564 Compilation
5 : INF560 Algorithmique parallèle et distribuée
6 : INF559 Architecture des ordinateurs
7 : INF580 Programmation par contraintes et programmation mathématique
8 : INF558 Théorie de l'information
9 : INF562 Géométrie algorithmique : de la théorie aux applications
10 : INF551 Computer-aided reasoning
11 : INF568 Cryptologie
12 : INF583 Principes et programmation des systèmes d'exploitation
13 : INF550 Conception et analyse des algorithmes
14 : INF549 Programmation en OCaml
15 : INF321 Les principes des langages de programmation
16 : INF570 Réseaux pairs à pairs
17 : INF586 Sécurité des réseaux
18 : INF554 Utilisation de l'aléatoire en algorithmique
19 : INF556 Modélisation des systèmes d'information
20 : INF585 Programmation en C++
21 : INF555 Geometric Modeling: Digital Representation and Analysis of Shapes
22 : INF441 Modal d'Informatique
23 : INF422 Composants d'un système informatique : Introduction à l'architecture des ordinateurs et à leurs systèmes d'exploitation
24 : INF565 Vérification
25 : INF557 Introduction aux réseaux
26 : INF581 Projet personnel en laboratoire
27 : INF423 Fondements de l'Informatique : logique, modèles, calculs

INF311 Introduction à l'informatique

Ce cours d'initiation s'adresse à des élèves de première année ayant peu ou pas de connaissances en informatique.

Une première partie aborde les bases de la programmation communes à la majorité des langages. On introduit la programmation objet, un des principaux paradigmes de programmation utilisés à l'heure actuelle. On verra comment cette approche facilite la conception de programmes; tout cela étant développé et implanté en Java.

Dans une deuxième partie, on aborde les différentes manières de représenter des données structurées comme les arbres, les graphes, les tables associatives ainsi que les algorithmes élémentaires les concernant.

Enfin, une dernière partie présentera certains outils conceptuels (automates finis, circuits booléens, etc.) qui permettent de modéliser des problèmes concrets et de s'assurer de la correction d'un programme.

Ce cours demande un certain travail personnel indispensable pour l'apprentissage de la programmation.

Responsable : Francois MORAIN
Période : Printemps
Nombre d'heures : 40
Crédits ECTS : 5
Dernière mise à jour : jeudi 05 avril 2012

INF431 Algorithmique et programmation

L'objectif de ce cours est de doter les élèves d'un solide socle de connaissances fondamentales en ce qui concerne la conception des algorithmes, la preuve de leur correction, l'analyse de leur complexité, et plus généralement la conception et la réalisation de systèmes logiciels.

L'informatique est d'une part une discipline scientifique à part entière, d'autre part un puissant outil, exploité par toutes les disciplines, y compris l'informatique elle-même.

Ce cours constitue une introduction approfondie aux notions fondamentales de l'informatique, aussi bien scientifiques que techniques. Il s'adresse à ceux qui s'intéressent à l'informatique pour elle-même comme à ceux qui s'y intéressent en tant qu'outil.

Ce cours enseigne d'abord l'algorithmique, c'est-à-dire la science de la résolution systématique des problèmes et de l'organisation efficace des données. L'accent est mis sur la conception des algorithmes et sur l'analyse de leur efficacité. Un petit nombre de concepts et de techniques fondamentaux, qui sont exploités dans de nombreuses situations, sont présentés.

De plus, ce cours est conçu comme un approfondissement de la pratique de la programmation. Le langage Java est utilisé tout au long du cours, pour les travaux dirigés sur machine et pour un projet de programmation réalisé en binôme. Ceci se double d'une réflexion théorique à propos de la programmation: on propose une introduction au test et à la preuve de programmes ainsi qu'une introduction à la programmation concurrente.

Ce cours est un pré-requis général pour accéder à tous les cours d'informatique en année 3 à l'exception du PA bio-informatique.

Niveau requis
En principe, il est nécessaire d'avoir suivi INF321 ou (INF311 et INF421).
Il n'est pas nécessaire d'avoir suivi INF422 ou INF423.

Modalités d'évaluation
Deux contrôles classants par écrit.
Un projet de programmation réalisé seul ou en binôme.

Responsable : Benjamin WERNER, François POTTIER
Période : Printemps
Nombre d'heures : 72
Crédits ECTS : 10
Dernière mise à jour : mercredi 20 juin 2012

INF421 Les bases de la programmation et de l'algorithmique

Initiation aux structures des données, à l'algorithmique et à l'analyse des algorithmes (amphis).
Perfectionnement à la programmation en Java (TDs).

Ce cours entend amener les élèves de niveau INF311 au niveau de fin de INF321, afin de pouvoir poursuivre le cursus d'informatique de l'Ecole. L'enseignement porte essentiellement sur les structures de données (graphes, listes, piles, files, arbres), sur l'algorithmique, et sur un minimum de théorie de la complexité. Ce cours est également l'occasion de se perfectionner en Java. Les TDs explorent en profondeur une partie du matériel presenté en amphi.

Niveau requis

INF311

Modalités d'évaluation
Un contrôle hors classement sur machine, un examen classant en salle (3h).

Responsable : Leo LIBERTI
Période : Automne
Nombre d'heures : 36
Crédits ECTS : 5
Dernière mise à jour : mardi 01 novembre 2011

INF564 Compilation

Le cours consiste à étudier et à écrire un compilateur d'un langage impératif simple, baptisé Pseudo-Pascal, vers l'assembleur MIPS. Il permet de comprendre l'étendue et le franchissement du fossé qui sépare langages de haut niveau et langages machine. Il permet également de découvrir des techniques et algorithmes non triviaux et de les exprimer dans un langage de très haut niveau, à savoir Objective Caml.

Niveau requis
Une journée de formation à la programmation en OCaml est organisée en septembre et est considérée comme un pré-requis pour le cours INF564. En d'autres termes, cette formation est obligatoire.

Modalités d'évaluation
Examen écrit.

Responsable : François POTTIER
Période : Printemps
Nombre d'heures : 36
Crédits ECTS : 4
Dernière mise à jour : vendredi 29 mars 2013

INF560 Algorithmique parallèle et distribuée

Le parallélisme et la distribution sont omniprésents en informatique, depuis le simple PC, déjà multi-coeur, jusqu'aux grilles de calculs, en passant par les cartes graphiques, les réseaux d'ordinateurs, les bases de données distribuées et les super-calculateurs.

Mettre en œuvre le parallélisme impose de repenser les applications, donne une nouvelle dimension à l'algorithmique, et nécessite de prendre en compte les paramètres clés de l'architecture cible. Ce cours propose une introduction approfondie aux concepts et techniques de l'algorithmique, de la programmation et de la sémantique parallèle, avec une mise en oeuvre conséquente.

Les TDs permettent d'expérimenter le parallélisme à travers deux grands modèles: les clusters de PCs avec JAVA RMI, et le parallélisme intensif sur cartes graphiques sous C/CUDA.

Niveau requis
Avoir suivi INF431 ou équivalent

Modalités d'évaluation
Par projet à l'oral

Responsable : Éric GOUBAULT
Période : Printemps
Nombre d'heures : 36
Crédits ECTS : 4
Dernière mise à jour : jeudi 28 mars 2013

INF559 Architecture des ordinateurs

Le but du cours est de comprendre la structure d'un processeur puis d'un système matériel complet, les contraintes qui ont abouti aux architectures actuelles, la programmation de ces architectures. Les étapes du cours sont: circuits combinatoires et séquentiels, architecture d'un microprocesseur RISC, programmation d'un microprocesseur en assembleur, système matériel (périphériques, interruptions, mémoire physique et virtuelle), processeurs superscalaires et à venir, programmation et performance.

Les principaux chapitres du cours:
- Représentation des nombres
- Circuits logiques
- Unité Arithmétique et Logique
- Notions de temps et de mémorisation
- Contrôle et jonction des composants
- Evolution des ordinateurs – Historique
- Un microprocesseur simple (LC2)
- Programmation d'un microprocesseur
- Système complet
- Les microprocesseurs actuels:
-- Pipeline
-- Prédiction de branchement
-- Caches
-- Exécution superscalaire
-- Multi-Coeurs

Niveau requis
La programmation d'un langage tel que Java ou C ou C++.

Modalités d'évaluation
- Examen écrit (3 heures)
- Projets optionnels

Responsable : Olivier TEMAM
Période : Automne
Nombre d'heures : 36
Crédits ECTS : 4
Dernière mise à jour : vendredi 14 octobre 2011

INF580 Programmation par contraintes et programmation mathématique

La programmation par contraintes consiste à poser un problème sous forme de contraintes entre plusieurs variables. Les différentes méthodes de résolution se distinguent par la manière de propager des valeurs impossibles pour certains variables. La programmation mathématique concerne le cas de variables numériques sous des contraintes numériques, souvent linéaires. Il existe de nombreux solveurs implémentés et ces formes de résolutions de problèmes sont appréciées en industrie.

Ce cours présente les fondements théoriques à la programmation par contrainte (CSP), et de la programmation mathématique. Il montre comment s'en servir pour résoudre des problèmes concrets. Il y aura des exercices théoriques, mais aussi pratiques utilisant une bibliothèque de CSP, et un langage de modélisation en programmation mathématique.

Modalités d'évaluation
20 % projet, 80 % examen écrit final

Responsable : Christoph DURR
Période : Hiver & Printemps
Nombre d'heures : 36
Crédits ECTS : 4
Dernière mise à jour : lundi 25 mars 2013

INF558 Théorie de l'information

La théorie de l'information répond à des questions diverses qui se posent par exemple dans les communications numériques quand on cherche à transmettre les messages de la façon à la fois la plus économique et la plus sûre.
Dans ce contexte, on peut se poser la question de déterminer le taux de compression ultime que l'on peut espérer ou le débit maximal avec lequel on peut transmettre de manière fiable sur un canal bruité.
Claude Shannon a donné une réponse magique à ces questions en proposant une manière de quantifier l'information d'un message et en montrant que le taux de compression ou que le débit de transmission maximal pouvaient tous deux s'exprimer en fonction de cette quantité d'information justement. Il a aussi proposé des techniques de compression ou de codage correcteur d'erreur (qui n'étaient absolument pas pratiques...) pour atteindre les limites mentionnées plus haut.
Nous expliquerons tout cela dans ce cours et nous verrons les réponses modernes à ces questions qui permettent elles d'atteindre de manière efficace les limites posées par Shannon.

Responsable : Jean-Pierre TILLICH
Période : Automne
Nombre d'heures : 36
Crédits ECTS : 4
Dernière mise à jour : jeudi 11 octobre 2012

INF562 Géométrie algorithmique : de la théorie aux applications

La géométrie algorithmique est une jeune discipline de l'informatique qui étudie d'un point de vue combinatoire et algorithmique les propriétés d'objets géométriques tels que nuages de points, arrangements, graphes géométriques, ou encore triangulations.

Ce cours propose une promenade au sein de cette discipline afin d'en illustrer la richesse sur le plan théorique et applicatif. Dans ce contexte, nous introduirons un éventail de problèmes issus du domaine, des plus classiques comme le calcul d'enveloppes convexes ou de triangulations de Delaunay, aux plus récents comme la reconstruction à partir de nuages de points, l'approximation de problèmes géométriques NP-difficiles, ou la localisation éfficace de points en grandes dimensions.

L'objectif du cours sera double : d'une part, mettre en relief l'élégance et la validité théorique des solutions proposées ; d'autre part, montrer leur potentiel au travers d'applications issues de domaines tels que l'informatique graphique, la robotique, l'apprentissage ou le traitement d'images.

Niveau requis
INF555

Modalités d'évaluation
Examen écrit + partiel ecrit et sur machine

Responsable : Luca CASTELLI ALEARDI, Steve OUDOT
Période : Printemps
Nombre d'heures : 36

Crédits ECTS : 4
Dernière mise à jour : vendredi 22 mars 2013

INF551 Computer-aided reasoning

Ce cours est une introduction à la formalisation des mathématiques, en théorie et en pratique, et à la résolution de problèmes logiques à l'aide d'ordinateurs.
Il revient sur les fondements théoriques des mathématiques, sous la forme de la Théorie des Ensembles et de ses alternatives plus computer-friendly, et il parcourt les techniques très diverses de la Déduction Automatique à la preuve interactive de théorèmes.
Les TD utiliseront l'assistant à la preuve Coq, notamment pour la formalisation de problèmes d'algèbre et de leur preuves.
Mais des SAT-solveurs, des SMT-solveurs ou des prouveurs automatiques pourront être utilisés pour illustrer d'autres techniques et applications.
Les étudiants bénéficieront d'initiative et de flexibilité sous la forme de mini-projets, ou de moins-mini-projets pour ceux qui décident d'approfondir ces thèmes dans l'EA MAT/INF571.

COURS:

Formalisation des problèmes et des théories mathématiques

- Rappels :
logique, modèles, arithmétiques de Péano et de Presburger, théorèmes de Church et Goedel.
- Théorie des ensembles :
Version naïve et paradoxe de Russel, axiomes de Zermelo-Fraenkel, indécidabilité et incomplétude.
Ordinaux et récurrence transfinie, axiomes du choix et du continu.
- Théorie des fonctions :
Lambda-calcul et paradoxe de Russel, théorie des types.
- Preuves constructives :
contenu calculatoire des preuves, versions constructives du choix, fondements de Coq.

Méthodes de démonstration automatique

- Dans le cas de théories décidables : SAT et SAT-modulo-Theory.
- Dans certaines théories: élimination des quantificateurs.
- Recherche de preuve généraliste : preuve en avant / en arrière, unification.
- Recherche de preuve comme principe de programmation : introduction à la programmation logique / ProLog.

TD sur machine:

Introduction à la pratique de Coq, exercices d'arithmétique, etc...
Extraction de programme à partir d'une preuve constructive : exemple d'un programme de tri.

SAT : Programmation d'un SAT-solveur basique. En Coq-même ou dans un autre language.
Projet en Coq ou programmation d'un prouveur dans un langage au choix.

Niveau requis
INF431 n'est pas un pré-requis.
INF423 est très conseillé, même s'il est possible d'acquérir rapidement les concepts nécessaires (formules logiques, modèles, preuves, fonctions calculables,...)

Modalités d'évaluation
Examen sur table (2 ou 3 heures), pour ceux qui prennent ce module comme cours.
Mini-projet, pour ceux qui prennent ce module comme EA.

Responsable : Stéphane LENGRAND
Période : Automne
Nombre d'heures : 36
Crédits ECTS : 4
Dernière mise à jour : vendredi 22 mars 2013

INF568 Cryptologie

Le but de ce cours est de se familiariser à la théorie et à la pratique des systèmes de chiffrement modernes.

On commencera par construire les primitives cryptographiques que sont le chiffrement et la signature. Ces primitives seront utilisées pour construire des protocoles plus complexe (authentification, échange de clefs, etc.) et plus proches de la vie courante. On présentera les systèmes de chiffrement modernes, qu'ils soient symétriques (DES, IDEA, AES) ou asymétriques (RSA). On analysera la sécurité de ces systèmes en les reliant à des problèmes difficiles (factorisation d'entiers, logarithme discret, etc.), preuves de sécurité de type OAEP.

Nous mêlerons théorie et pratique autant que faire se peut. Les algorithmes et protocoles seront programmés en Java ou dans un langage de haut niveau alternatif.

Niveau requis
Les cours du premier sémestre : INF557 et INF558, ou cours similaires.

Modalités d'évaluation
Projets.

Responsable : Francois MORAIN
Période : Printemps
Nombre d'heures : 36
Crédits ECTS : 4
Dernière mise à jour : jeudi 05 avril 2012

INF583 Principes et programmation des systèmes d'exploitation

Introduction aux principes des systèmes d'exploitation, à leur structure interne, et à leur interface de programmation.

Ce cours porte sur la conception des systèmes d'exploitation et la programmation système (en C). Étude générale de la concurrence et des communications entre processus, et analyse des mécanismes d'abstraction et de virtualisation. Étude des couches basses du noyau (processus, threads, périphériques, gestion des interruptions, ordonnancement, mémoire virtuelle, protection) et des services de plus haut niveau (contrôle des entrées-sorties, systèmes de fichiers, politique de sécurité, édition de liens dynamiques). Ouverture vers les machines virtuelles, les systèmes pour architectures parallèles à mémoire partagée et les systèmes embarqués (contraints en ressources ou temps-réel).

Le cours repose en partie sur les acquis de INF422, ou une introduction équivalente à l'organisation des systèmes informatiques.

Niveau requis
INF422 est recommandé, ou compétences comparables sur l'utilisation et la programmation du Shell.

Le langage de programmation utilisé pour décrire et programmer les composants du système est le C (pas le C++). Un niveau élémentaire en C est suffisant (en complément d'une bonne maîtrise des principes des langages de programmation), mais les étudiants sont encouragés à en apprendre davantage au fur et à mesure (supports fournis).

Modalités d'évaluation
Examen oral avec mini-projet sur la base de l'un des sujets de TD.

Responsable : Albert-Henri COHEN
Période : Printemps
Nombre d'heures : 36
Crédits ECTS : 4
Dernière mise à jour : lundi 25 mars 2013

INF550 Conception et analyse des algorithmes

Ce cours porte sur des techniques avancées dans la conception et l'analyse des algorithmes. Le cours commence par un parcours rapide de quelques uns des paradigmes principaux de l'algorithmique, notamment flots et couplages et programmation linéaire. Il revient ensuite rapidement sur la NP-complétude et notamment sur l'importance des réductions polynomiales. Au delà de l'analyse de pire cas, on envisage plusieurs approches possibles de l'analyse d'algorithmes, d'une part au travers des notions de complexités: pire-cas, en moyenne, amortie, lissée, ou paramétrique, et d'autre part au travers de mesures de qualités de sortie: optimalité, facteur d'approximation pour l'optimisation, de compétitivité pour l'algorithmique on-line. Ce sera l'occasion de présenter entre autres les notions de potentiel, de

noyau, de largeur arborescente.

Niveau requis
Les prérequis minimaux sont quelques éléments d'algorithmique (manipulations de tableaux, listes, arbres; tris et parcours fondamentaux) et de complexité des algorithmes (bornes supérieures et inférieures à la complexité dans le pire cas pour les tris), et une bonne familiarité avec l'outil mathématique (preuve par induction, etc).

Avoir suivi INF423, INF431 ou des cours d'informatique de niveau équivalent est utile.

Modalités d'évaluation
Examen sur table à l'issue du cours.

Responsable : Gilles SCHAEFFER
Période : Automne
Nombre d'heures : 36
Crédits ECTS : 4
Dernière mise à jour : mardi 20 mars 2012

INF549 Programmation en OCaml

Ce cours est une introduction / mise à niveau à la programmation en OCaml, en pré-requis à

Parcours Conception des Systèmes Informatiques (obligatoire)
Parcours Transverse (obligatoire)
INF564 (obligatoire)
INF551 (fortement conseillé)
INF565 (fortement conseillé)

Ce cours aura lieu le lundi 1er octobre (amphi de 8h30 à 10h00 en PC24, TD de 10h15 à 12h15 et de 18h00 à 19h00 en salle info 33).

La page du cours

Modalités d'évaluation
Aucune.

Responsable : FILLIATRE
Période : Automne
Dernière mise à jour : vendredi 22 mars 2013

INF321 Les principes des langages de programmation

Ce cours a pour objectif de consolider les connaissances des élèves en programmation, leur offrant un panorama
des principaux paradigmes existants: impératif (C, Java), fonctionnel (OCAML), orienté-objets (Java, OCAML) etc.
Les concepts de ces langages de programmation sont introduits en en donnant une sémantique mathématique précise.

Ce cours s'adresse aux élèves de première année ayant déjà acquis quelques connaissances en informatique avant leur entrée à l'École.

Il présente les grands principes d'organisation des langages de programmation, langages dont la conception, depuis le milieu du XXe siècle, a profondément changé notre rapport au langage en général, aux machines et à la complexité.

Ce cours permettra aux élèves de systématiser les connaissances sur les langages de programmation qu'ils ont souvent acquises empiriquement en programmant et il leur donnera les outils conceptuels nécessaires pour aborder de nouveaux langages dans la suite de leur cursus à l'École et au-delà.

Niveau requis
Une aisance minimale en programmation, impérative ou fonctionnelle, ainsi qu'en algorithmique de base, est requise.

Modalités d'évaluation
Les étudiants sont évalués par un écrit de 2h, en fin de cours.

Responsable : Éric GOUBAULT
Période : Printemps
Nombre d'heures : 40
Crédits ECTS : 5
Dernière mise à jour : jeudi 28 mars 2013

INF570 Réseaux pairs à pairs

L'objectif de ce cours est de comprendre les principes des systèmes distribués à large-échelle, tels que les systèmes pair-à-pair.

Partant d'une présentation des applications pair-à-pair existantes nous présenterons les problèmes que de telles applications doivent résoudre (e.g., routage, choix des paires et du contenu, impact du protocole de transport sur la qualité de service et d'experience, etc.). Ensuite, nous présenterons différentes solutions à ces problèmes, en considérant aussi bien les réseaux non-structurés (e.g., diffusion naïve par inondation, protocoles épidémiques) que les réseaux structurés (e.g., tables de hachage distribuées).

Le cours est structuré autour des applications les plus importantes et populaires (e.g., Gnutella, Bittorrent, Skype, Kademlia, etc.), qu'il attaque de plusieurs point de vue (du plus theorique au plus pratique). Completent le cours un ensemble de lectures (generalement, papiers de recherche) et un projet personnel, pour permettre de mieux comprendre les réseaux pair-à-pair via l'évaluation de leur performances.

1. Introduction
2. Skype
3. P2P-TV
4. BitTorrent
5. LEDBAT/uTP
6. DHT and lookup
7. Topology management and network awareness

Modalités d'évaluation
La note du cours sera la moyenne
* d'une soutenance orale du project, évaluée comme d'habitude par une note entre A et F.
* d'un examen écrit (format QCM) sur l'ensemble du cours

Les meilleurs projets pourraient donner lieu à une soumission du travail à une conference du secteur (au dela du cours; travail supplementaire à prevoir). L'acceptation du papier donnerait la possiblité à l'étudiant de presenter ses travaux à la conference.

Responsable : Dario ROSSI
Période : Automne
Nombre d'heures : 36
Crédits ECTS : 4
Dernière mise à jour : vendredi 22 mars 2013

INF586 Sécurité des réseaux

L'objectif de ce cours est de comprendre et d'analyser les outils qui permettraient de sécuriser internet et les réseaux locaux.

Nous étudierons les vulnérabilités de chaque couche du réseaux, du hacking de réseau WiFi à l'intrusion et l'écoute de liaisons TCP et applicatives en passant par le détournement des connexions liés aux faiblesses des protocoles de routage.

Une fois ces menaces maîtrisées, nous examinerons un ensemble de protocoles et d'architectures actuellement utilisés pour la sécurisation d'Internet et des réseaux locaux dans le but d'obtenir une infrastructure de communication robuste. Cela comprend l'étude et la compréhension des protocoles et techniques telles que SSL, SSH, VLAN, VPN, WPA2, Kerberos...

Il poursuit deux buts. Le premier est d'intégrer les principes généraux liés aux attaques et à la sécurisation

du réseau afin de pouvoir plus facilement réagir aux nouvelles attaques et comprendre les nouveaux dispositifs de sécurité. Le second est d'avoir une vue exhaustive de l'ensemble des risques et des solutions en matière de sécurité des réseau et ainsi d'être capable d'analyser quels sont les risques associés à un cas de figure particulier et de proposer une solution adaptée.

Le module est constitué d'enseignements magistraux ainsi que de TP réalisé sur les portables de étudiants afin de mettre en place des solutions de sécurité pour des problèmes pratiques.

Un fois le module suivi, un étudiant sera capable de :
effectuer des audits de sécurité des infrastructures réseaux des entreprises
de proposer des solutions de sécurisation
d'installer ou de faire installer ces solutions

Niveau requis
INF557 introduction aux réseaux ou équivalent

Modalités d'évaluation
Le cours est évalué par:
- des comptes-rendu de TP
- un examen final où les étudiants doivent proposer et configurer des solutions pour des cas pratiques précis.

Responsable : Julien CERVELLE
Période : Printemps
Nombre d'heures : 36
Crédits ECTS : 4
Dernière mise à jour : mardi 03 juillet 2012

INF554 Utilisation de l'aléatoire en algorithmique

Il y a plusieurs manière d'utiliser l'aléatoire en informatique. On peut étudier le comportement d'un processus informatique (programme, protocole, automate, …) face à un modèle aléatoire, ou encore construire des générateurs d'instances aléatoires (séquences, arbres, graphes, …) à des fins applicatives. Une autre approche, celle de ce cours, considère l'aléatoire comme une nouvelle ressource permettant de résoudre plus efficacement certains problèmes. Par exemple, on ne sait pas générer un nombre premier sans tirer au sort, ou encore la congestion dans les réseaux ne peut pas être évitée sans un protocole de routage en partie aléatoire.

De manière contre-intuitive, prendre des décisions au hasard permet de concevoir des algorithmes plus simples et souvent plus efficaces que leurs analogues déterministes. Les domaines et les applications sont vastes. Ce cours fournit une présentation accessible à la plupart de ces derniers et de leurs idées centrales. Chacune de ces idées sera présentée à travers des exemples simples et motivés.

Nous commencerons par acquérir un certain nombre de méthodes et outils à travers des problèmes

fondamentaux de l'informatique, incluant le test de primalité, la résolution de problèmes SAT, l'algorithmique de graphe ou encore certains problèmes algébriques.
Puis nous étudierons les algorithmes de streaming, où l'utilisation de l'aléatoire est cruciale afin de réduire l'espace mémoire de ces algorithmes qui manipulent des données de taille gigantesque. L'étude en complexité de ces algorithmes est liée au modèle de la complexité de la communication. Ce modèle a de multiples applications, comme dans l'optimisation des circuits imprimés.
L'utilisation de l'aléatoire permet aussi de revisiter et d'enchérir la notion de preuve, comme nous le verrons lors de l'étude des preuves interactives probabilistes.
Enfin, l'étape ultime consiste à remplacer l'utilisation de phénomènes probabilistes par l'utilisation de phénomènes quantiques. Sans connaissance a priori de la physique quantique, il est possible d'aborder cette dernière sous le regard neuf de l'informatique. Cette interprétation des paradoxes quantiques a donné lieu à des protocoles cryptographiques et à des algorithmes, n'ayant aucun analogue probabiliste.

Les domaines abordés seront :
- Classes de complexité probabilistes
- Techniques d'amplification d'erreur
- Principe de hâchage
- Marches aléatoires en algorithmique
- Méthodes de dérandomisation
- Algorithmes de streaming
- Complexité de la communication
- Preuves interactives
- Ouverture sur l'informatique quantique

Niveau requis
Ce cours ne nécessite aucun prérequis particulier.
En revanche, un minimum de connaissances algorithmiques (principalement) et de probabilités discrètes (secondairement) peuvent aider.
Tous les outils probabilistes ainsi que les notions (élémentaires) de physique quantique nécessaires seront introduits durant le cours.

Ce cours est complémentaire des cours Conception et analyse des algorithmes et Randomization in Computer Science: Games, Networks, Epidemic and Evolutionary Algorithms, mais peut être suivi indépendamment.

Modalités d'évaluation
L'évaluation consistera en un travail personnel (rédaction de notes de cours, DM ou projet), ainsi qu'un examen écrit. Elle sera précisément déterminée lors du premier cours.

Responsable : Frédéric MAGNIEZ
Période : Automne
Nombre d'heures : 36
Crédits ECTS : 4
Dernière mise à jour : samedi 23 mars 2013

INF556 Modélisation des systèmes d'information

Ce cours a pour objectif de présenter les principales étapes d'une démarche de modélisation d'un système d'information d'entreprise. Il est organisé autour de l'apprentissage d'un processus de modélisation de système d'information. Le cours traite notamment de la manière dont on peut dégager l'architecture de l'environnement (i.e. la spécification des usages des utilisateurs) et fonctionnelle (i.e. l'organisation des fonctionnalités internes) d'un système d'information. Il s'appuie sur le langage UML (Unified Modelling Language) pour représenter ces différentes vues architecturales.

1 - Introduction à la modélisation - Eléments de contexte
2 - Introduction à la modélisation - Cadre d'architecture d'entreprise
3 - Modélisation opérationnelle - Environnement d'un système d'information
4 - Modélisation opérationnelle - Modélisation des besoins
5 - Modélisation opérationnelle - Contextes opérationnels
6 - Modélisation opérationnelle - Usages d'un système d'information
7 - Modélisation opérationnelle - Scénarios d'usage
8 - Modélisation fonctionnelle - Fonctionnalités
9 - Modélisation fonctionnelle - Fonctionnements et modes de fonctionnement

Niveau requis
INF431

Modalités d'évaluation
Projet de synthèse à réaliser en binôme (modélisation opérationnelle et fonctionnelle d'un logiciel à choisir par le binôme)

Responsable : Daniel KROB
Période : Automne
Nombre d'heures : 36
Crédits ECTS : 4
Dernière mise à jour : mardi 02 avril 2013

INF585 Programmation en C++

Le but de ce cours est de donner une connaissance pratique de C++ aux élèves ayant un niveau limité en informatique. L'environnement de travail est Unix (Linux).

Syntaxe et utilisation basique du C++. Introduction à la mémoire et aux pointeurs. Usage basique de la Librairie Standard (Standard Template Library, STL). Polymorphisme de fonctions et opérateurs. Quelques mots sur l'architecture logiciel. Introduction à la programmation orientée objet. Héritage de classe. Techniques de développement et de debugging sous Linux. Une sélection de thèmes avancés entre : usage avancé des pointeurs, héritage virtuel, usage avancé de la Librairie Standard, programmation type "template". Ce cours suppose la connaissance d'au moins un autre langage de programmation générique

(Java, C, Fortran, Visual Basic ...).

La "théorie" est introduite au cours des 4 premières séances. Les 5 dernières séances sont dédiées à un projet qui implique toute la classe: un logiciel "grande taille" est conçu, planifié et implémenté en équipe. L'evaluation est faite sur la qualité du logiciel implémenté.

Niveau requis
Apprendre le C++ nécessite un fort intérêt à l'architecture de l'ordinateur et au système d'exploitation, la principale raison étant la possibilité d'accès direct à la mémoire. C'est pour cela qu'un minimum de pré requis en informatique est nécessaire. En particulier les élèves devront connaître les outils basics Unix : ls, cp, mv, cd, mkdir, rmdir. On utilisera Emacs comme éditeur de texte et le compilateur GNU C++.

Modalités d'évaluation
Les modalités d'examen changent chaque années, mais la partie plus importante de l'évaluation est dans le projet, qui implique toute la classe. Chaque année on choisit un sujet logiciel et 2 "responsables architectes" du code, qui forment et coordonnent des équipes d'élèves pour produire un logiciel suffisamment complexe. Une partie auxiliaire de l'évaluation se fait par examen soit à l'ordinateur soit oral.

Responsable : Claudia D'AMBROSIO
Période : Printemps
Nombre d'heures : 36
Crédits ECTS : 4
Dernière mise à jour : lundi 25 mars 2013

INF555 Geometric Modeling: Digital Representation and Analysis of Shapes

Représentation numérique et analyse de formes

La compréhension et la manipulation des formes est au cœur de l'interaction humaine avec le monde qui nous entoure. Comme les enfants, nous arrivons vite à reconnaître et classifier les objets à partir de leurs formes, et nous sommes aussi poussés à créer nos propres formes par le biais de dessins et croquis.

En même temps, apprendre aux ordinateurs à générer et manipuler des formes s'est révélée être une tâche très complexe, car ils sont traditionnellement conçus et considérés comme des «calculateurs», plutôt que comme des machines dotées de sens «artistique» et de perception.

Néanmoins, d'énormes progrès ont été faits dans cette direction, permettant le développement de domaines tels que la conception assistée par ordinateur (dans la réalisation de produits manufacturés), les interfaces graphiques, les jeux vidéos, le graphisme et les images de synthèse, entre autres.

Dans ce cours nous allons introduire les notions fondamentales pour la création et la manipulation numérique de formes. On commencera par la génération et la représentation de courbes lisses en 2d, en utilisant les B-splines et les courbes de Béziers. On passera par la suite aux différentes techniques pour la représentation de formes en 3d, avec un accent particulier sur les maillages triangulaires et les méthodes

associées. En même temps, nous introduirons des méthodes pour l'analyse de formes, permettant en particulier le calcul de similarités et correspondances entre formes (comment établir des correspondances entre points se trouvant sur une forme donnée).

Nous traiterons aussi d'autres sujets, tels que :
1) Représentations paramétriques et implicites de courbes
2) formes polaires et algorithme de subdivision de de Casteljau
3) courbes et surfaces de subdivision
4) Représentations et algorithmes pour les nuages de points
5) Maillages triangulaires et NURBS
6) Recalage rigide
7) Paramétrisation de formes et placage de textures
8) Méthodes spectrales pour l'analyse de formes

La plupart des séances seront assurées en anglais.

Modalités d'évaluation
Oral + soutenance de mini projet

Responsable : Luca CASTELLI ALEARDI
Période : Automne
Crédits ECTS : 4
Dernière mise à jour : mercredi 27 mars 2013

INF441 Modal d'Informatique

Attention, pour le choix du modal, vous devez choisir une période. Ainsi si vous désirez suivre le modal programmation efficace par exemple, celui-ce ne se déroulant qu'en période 4, vous devez choisir modal informatique période 4. Une fois affecté, le responsable du modal vous enverra un mail vous demandant votre choix d'option (web, réseau ou programmation efficace en période 4) et votre choix sera automatiquement accordé.

Le modal informatique introduit différents domaines où l'expérience et la pratique interviennent de façon cruciale, de façon à proposer une solution à certains problèmes difficiles.Quand on accède à une page page, comment les données (images, texte, …) arrivent-elles depuis le serveur web jusqu'au navigateur via Internet ? Quels protocoles permettent au réseau mondial de fonctionner? Autour de quels algorithmes et paradigmes ces protocoles sont-ils construits? Envoyer de l'information au travers d'un réseau, potentiellement dynamique, tel qu'Internet, et ce d'une manière optimale dépend non seulement de la topologie du réseau mais aussi de contraintes physiques ce qui rend difficile mais passionnant l'élaboration d'algorithmes adaptatifs.Facebook, twitter, flickr ... sont autant d'applications utilisées quotidiennement par des millions de personnes depuis leur ordinateur, leur téléphone ou leur tablette. Comprendre et réaliser une telle application est-il difficile ? L'imagerie numérique a remplacé en moins de 10 ans l'argentique et tout un nouveau monde autour de la retouche d'image, de la navigation virtuelle ou de l'analyse d'image comme la reconnaissance faciale se développe naturellement. Enfin, s'assurer de la correction d'un

programme devient indispensable dans de nombreuses applications critiques comme par exemple pour les logiciels embarqués dans les voitures, les avions ...
À toutes ces thématiques, le modal informatique tente de donner des réponses et d'illustrer des aspects plus fondamentaux des autres cours d'informatique du département.

Le département offre différentes options, chacune ayant son propre numerus clausus.

Période 1 (Sept - Oct.): Pas d'option proposée
Période 2 (Oct. - Jan): Applications pour téléphone
Période 3 (Jan - Avr.): Web - Bioinformatique - Image
Période 4 (Avr - Jui): Web - Programmation Efficace - Réseau

Développement d'applications pour téléphones et tablettes (Resp. Dominique Rossin)

Prérequis: Aucun

Vous pouvez suivre cette option sur la période Oct-Jan.

Demandez-vous combien d'applications sur ordinateur vous utilisez actuellement et combien de temps vous passez
sur des applications mobiles sur votre téléphone ou votre tablette. Si vous repensez à cette question en vous projetant
3 ans, 5 ans ou 7 ans en arrière vous réaliserez que nous sommes actuellement en train de vivre un changement important
et rapide dans notre mode de vie.

Dans cette option, nous aborderons la problématique de développer des applications pour téléphone mobile ou tablette.
Afin d'assurer une compatibilité maximale, nous aborderons ce problème de manière similaire à l'option programmation Web
du modal Informatique. Ainsi, dans un premier temps nous verrons comment réaliser une application Web puis comment
transformer cette application en une application pour téléphone mobile de manière automatique. Il sera ainsi possible
d'utiliser les différents capteurs d'un téléphone comme le GPS ou l'appareil photo et de créer des applications les utilisant.
Une description plus complète de cette option sera donnée lors de la réunion pour le choix des modal et sur la page Web du cours.

Développement d'applications Web. (Resp. Olivier Serre, Dominique Rossin)

Prérequis: Aucun

Vous pouvez suivre cette option sur la période de votre choix (Jan-Avr / Avr-Jui)

En 2011, le nombre de sites Web s'élevait à environ 155.000.000 alors qu'il n'en existait que 54.000.000 en 2004. De plus, ces sites proposent de plus en plus de services personnalisés suivant l'utilisateur : agrégateurs, espaces de travail partagé, sites communautaires ou encore blogs en sont des parfaits exemples. Cette nouvelle donne a vu se développer en parallèle des technologies adaptées pour le développement de tels sites ou devrait-on dire actuellement de telles applications.

Ce cours a pour objectif d'aborder d'un point de vue pratique les problèmes liés au développement de ces applications. Les techniques abordées seront les suivantes :

Programmation objet en PHP.
Introduction aux bases de données à travers MySQL.
Sécurité des applications, cartes et géolocalisation, javascript, Ajax.

Ce cours sera en majeure partie composé de TDs en salle machines, les élèves devant réaliser à terme un projet comme le développement d'une application Web permettant la gestion dynamique d'une bibliothèque (clients, stock, réservations, emprunts, rendus, etc.), un petit site d'hébergement de blogs, de binet, un site collaboratif ou tout autre application du même genre au choix...

Les TDs seront en plus l'occasion de découvrir par la pratique quelques notions-clé de l'informatique contemporaine, couramment employées dans le monde industriel.

Programmation efficace (resp. Christoph Dürr)

Prérequis : INF 311-421 ou INF 321, INF 431 conseillé

Cette option n’est proposée qu’en période 4 (Avr-Jui)

Le Modal programmation efficace a deux buts. Apprendre à implémenter un programme rapidement et trouver l'algorithme et l'implémentation la plus rapide pour un problème. Ce cours va développer des qualités de programmeurs qui seront demandées pour certains postes d'informaticien (par exemple Google).

L'idée est que les connaissances en génie logiciel ne peuvent être bien assimilées seulement après une certaine expérience en programmation. Concrètement on présentera une grande quantité d'algorithmes aux étudiants pour des problèmes combinatoires, de graphe et de géométrie algorithmique.

En parallèle les étudiants vont implémenter ces algorithmes et résoudre des problèmes du concours de programmation ACM (ICPC). On fera également des exercices pour développer le travail en groupe et la lecture de code source.

Réseaux (resp. Thomas Clausen)

Prérequis : Aucun

Cette option n’est proposée qu’en période 4 (Avril - Juillet)

Le but de ce modal est d'aborder la résolution de trois questions:

Comment écrire des applications utilisant le réseau, comme, par exemple, des outils de partage de fichier, des jeux en ligne voire un serveur web?

Nous allons étudier les primitives de programmation, les contraintes et principes nécessaires au développement de systèmes distribués ainsi que des considérations élémentaires concernant les algorithmes répartis utilisés par des logiciels tels Skype et IRC;

Comment Internet fonctionne-t-il vraiment?

Nous regarderons les protocoles qui interviennent dans la communication entre deux ordinateurs sur Internet, qui assurent l'administration du réseau et qui font en sorte que, où que l'on soit sur la planète, on accède à www.michelleobama.com : routage, DNS, ... Nous considérerons d'une part l'algorithmique sous-jacente qui fait fonctionner Internet ainsi que la manière dont ces derniers interviennent au niveau des protocoles.

Quelles sont les technologies derrière les substantifs "routeur", "commutateur (switch)", "répéteur (hub)", IPv6, VPN, etc?

Ce modal consiste en un petit nombre d'amphis de fond et une sélection d'amphis sur diverses technologies à choisir parmi plusieurs suite à une délibération entre enseignants et élèves. Le travail des élèves consiste, par groupe de deux ou trois, en la réalisation d'un projet durant les TD : un réseau ad-hoc sans fils entre mobiles et portables, une application distribuée de partage de fichiers, un serveur web distribué, un programme de discussion...

Imagerie numérique et Web (resp. Renaud Keriven)
Prérequis : Aucun

Cette option n’est proposée qu’en période 3 (Jan - Avr)

Aujourd’hui, les images ne sont plus uniquement un bien de consommation: nous en produisons tous à volonté.
Nous leur découvrons continuellement de nouvelles possibilités: se promener virtuellement dans la rue (Google Streetview) ;
naviguer en 3D dans ses propres photos (Microsoft PhotoSynth) ; y retrouver automatiquement ses amis (reconnaissance de visage de
Google Picasa) ; etc.

Dans ce modal, nous nous intéresserons plus particulièrement aux applications embarquées sur smartphone ou tablette ainsi qu'aux
applications web c'est à dire dans votre navigateur.
Pour se faire, nous présenterons à la fois les techniques usuelles de traitement photographique d'images 2D, 3D ainsi
que les problématiques liées au Web.

Ce modal sera l'occasion de développer un projet en groupe de 2 ou 3. Des exemples de réalisations possibles respectant
les centres d'intérêts de chacun ainsi que votre niveau sont:

Morphing de visage. À partir de deux photographies de visage, une animation est créée transformant le premier visage en le second.
StreetView Campus. Utiliser le GPS de votre mobile pour avoir une vue immersive du campus de l'endroit ou vous vous trouvez.
Déformation de visages au toucher. Glisser votre doigt sur votre smartphone et déformez à envie des visages.
Retouche Photo. Le sujet est large: d'Instagram à Photoshop réalisez votre logiciel de retouche pour votre navigateur ou votre smartphone.
HDR. À partir d'un ensemble de photos exposées différement, reconstruisez des images en ultra haute définition colorimétrique.
Reconnaissance faciale. Retrouvez un visage dans un ensemble de photos.
Scan de visage. Construisez un modèle 3D de votre visage à partir d'une photo.
Jeux réseaux 3D multijoueurs

Attention, ce modal n'est dispensé qu'en période 3 (Jan-Avr). Si vous désirez le suivre, choisissez modal informatique
pour cette période et vous choisirez votre option auprès du département d'informatique, celle-ci vous sera automatiquement accordée.

Pour tout renseignement, contactez Renaud Keriven ou Dominique Rossin

Bio-informatique (resp. Jean-Marc Steyaert)
Prérequis : Aucun

Les expérimentations en biologie fournissent un grand nombre d'informations. Ces informations locales permettent la reconstruction de structures compliquées soit de par leur taille, soit de par leur architecture. Le traitement informatique à grande échelle de ces données permettra ensuite de construire des modèles descriptifs ou explicatifs de phénomènes biologiques.

De nombreux logiciels d'usage courant sont élaborés à partir de méthodes programmatiques simples que l'on apprendra à étendre sur des exemples réels. On pourra traiter par exemple: recherche de gênes pathologiques, séquençage haut débit, reconstruction de génomes.

Ce modal peut être vu de deux manières différentes :
1- un ensemble de projets de programmation dans un domaine autre que l'informatique
2- une introduction concrète à la bioinformatique.

Modalités d'évaluation
Les différentes options du Modal seront ponctuées par la réalisation d'un projet qui composera la note

finale de ce module à l'exception du Modal Programmation Efficace qui propose un examen.

Responsable : Thomas CLAUSEN, Dominique ROSSIN, Christoph DURR, Renaud KERIVEN, Jean-Marc STEYAERT
Période : Hiver & Printemps
Nombre d'heures : 36
Crédits ECTS : 5
Dernière mise à jour : mercredi 30 mai 2012

INF422 Composants d'un système informatique : Introduction à l'architecture des ordinateurs et à leurs systèmes d'exploitation

On aborde de façon simple et synthétique un aspect important de l'univers informatique: l'architecture matérielle et logicielle des ordinateurs. Ces concepts et interfaces sont importants pour appréhender le fonctionnement des systèmes informatiques, de l'embarqué grand public aux plus grands réseaux.
Ce cours est adapté au niveau des élèves ayant suivi INF321 ou INF421. Il est ouvert aussi pour les élèves n'ayant suivi qu'INF311 mais prêts à fournir un travail un peu plus important sur les premières semaines.

Niveau requis
INF311 ou INF321

Responsable : Albert-Henri COHEN
Période : Automne
Nombre d'heures : 36
Crédits ECTS : 5
Dernière mise à jour : jeudi 19 mai 2011

INF565 Vérification

1) Initiation aux problèmes de vérification de logiciels : limitations fondamentales (calculabilité), principales approches, lien avec la compilation...
2) Réalisation pratique (projet) d'un logiciel de vérification ou d'un logiciel vérifié formellement.

De nos jours, de nombreux systèmes sont contrôlés par des logiciels de plus en plus complexes. De plus, les conséquences d'erreurs logicielles peuvent se révéler catastrophiques, dans le cas de systèmes embarqués (par exemple, dans le domaine aéronautique ou dans le secteur médical) ou de systèmes manipulant des informations sensibles (par exemple dans le domaine bancaire). La conception de systèmes matériels fait face à des difficultés similaires. Afin de maîtriser cette complexité on ne peut plus

se contenter de méthodes manuelles ou heuristiques.

Cet EA a pour objectifs :

- d'étudier comment un programme critique peut être garanti "sans bugs" grâce à des outils logiciels spécialisés
- de mettre en oeuvre ces techniques dans la réalisation d'une application embarquée et prouvée

Plus précisément, cet EA comporte une introduction aux techniques de vérification telles que :

- l'analyse statique de programme / l'interprétation abstraite
- le model-checking (à état explicite ou implicite)
- la résolution d'obligations de preuves en logique de Hoare
- la preuve interactive dans des logiques de spécification.

Cet EA donne les clefs de compréhension de ces techniques et surtout leur mise en pratique par l'usage d'outils logiciels tels que Coq, Why/Alt-Ergo, Astrée ou Java pathfinder...

Les travaux pratiques commenceront par le traitement de quelques problèmes simples de sécurité informatique et évolueront, au fur et à mesure des séances, vers des projets. Ceux-ci pourront tourner autour de la conception d'un petit interpréteur abstrait, de la certification d'un petit compilateur pour un mini-langage fonctionnel, ou d'autres projets plus individualisés selon les envies des élèves.

Remarque : cet EA fusionne les modules INF563 "Analyse statique de programmmes" et INF565 "Systèmes de preuve", avec un côté applicatif renforcé.

Niveau requis
Il est très fortement recommandé de savoir programmer en OCaml (il y a une remise à niveau au mois de septembre).

Il est également souhaitable (mais pas obligatoire) d'avoir des notions de raisonnement automatique par ordinateur, de logique ou encore de compilation.

Modalités d'évaluation
Projet, typiquement réalisation d'un logiciel prouvé en Coq ou programmé en OCaml, avec remise des listings et soutenance orale

Responsable : David MONNIAUX, Xavier RIVAL
Période : Printemps
Nombre d'heures : 36
Crédits ECTS : 4
Dernière mise à jour : jeudi 28 mars 2013

INF557 Introduction aux réseaux

Ce cours aborde la notion de réseau informatique dans une démarche descendante : comment fonctionne un programme (par exemple, un serveur sur la toile,
un système de dialogue ou Skype) qui doit communiquer avec un autre programme ?
Comment la résistance aux pertes est assurée ? Que se passe-t-il quand un paquet est acheminé à travers Internet ?
Quels sont les algorithmes qui permettent à une carte réseau de fonctionner ? Et pourquoi l'architecture des réseaux est-elle constituée de protocoles empilés ?

Ce cours présente une approche descendante, abstraite et architecturale, des réseaux informatiques. Entre autres sujets, nous allons étudier :

les architectures de réseaux
des algorithmes distribués
les protocoles de transport et de routage
les protocoles de liaison
les communications de groupe

Niveau requis
INF321 ou INF311+421 sont requis.
Une bonne dose de curiosité est recommandée.

Responsable : Thomas CLAUSEN
Période : Automne
Nombre d'heures : 30
Crédits ECTS : 4
Dernière mise à jour : jeudi 12 mai 2011

INF581 Projet personnel en laboratoire

Cet EA donne l'opportunité
- soit d'approfondir l'un des 3 cours choisis pour la période 2, de manière encadrée par l'enseignant ou l'un de ses collègues ;
- soit d'aborder un domaine nouveau que l'on a pas pu choisir en tant que cours, par une démarche plus pro-active de la part de l'étudiant ;
- soit éventuellement de valoriser, dans la scolarité de l'étudiant, un projet personnel qui resterait sinon un "hobby" hors-cursus.

Dans tous les cas, il s'agit d'un projet
- dans les sciences informatiques
- avec une dimension de recherche
- encadré par un enseignant ou un chercheur du campus

- cohérent avec le projet de formation et le projet professionnel de l'étudiant.

En comparant par exemple au projet programmation d'INF431 ou au PSC de 2ème année :
- le projet d'EA peut etre fait individuellement ou en binome ;
- une liste de sujets ne sera pas nécessairement proposée : c'est à l'étudiant de se demander quelles sont les choses qui l'intéressent, et d'aller voir les personnes du département qui sont les plus expertes dessus ;
- il ne s'agit pas nécessairement de programmation : pour des thèmes plus théoriques, cela pourra être l'étude d'une approche à un problème particulier avec une part de travail bibliographique.

De fait, le projet devra impliquer l'abord d'au moins un article de recherche et débouchera sur l'élaboration d'un rapport scientifique (et non pas d'activité) et une soutenance orale, devant les responsables de l'EA (informaticiens mais pas nécessairement spécialistes du sujet).

Celle-ci pourra impliquer une démonstration, si un outil a été programmé.

Responsable : Stéphane LENGRAND, Olivier BOURNEZ
Période : Printemps
Nombre d'heures : 36
Crédits ECTS : 4
Dernière mise à jour : vendredi 22 mars 2013

INF423 Fondements de l'Informatique : logique, modèles, calculs

Ce cours présente les fondements de l'informatique en tant que science. Si l'idée d'utiliser des machines pour effectuer des calculs est ancienne, c'est dans les années 30 que les travaux d'Alan Turing, Alonzo Church, Kurt Goedel et d'autres ont posé les bases de ce qui allait devenir l'informatique que nous connaissons aujourd'hui.

Leurs travaux ont révélé que le raisonnement et le calcul sont intimement liés et ces bases doivent donc se comprendre dans la tradition, plus ancienne, de la logique et des fondements des mathématiques, de Peano à Zermelo en passant par Hilbert et bien d'autres. Il est remarquable que ces bases sont toujours d'actualité, malgré les progrès technologiques spectaculaires.

Alors que d'autres cours montrent comment programmer, on explicite ici le cadre de ce qui est faisable, en termes
- de calculabilité : certains problèmes ne peuvent pas être résolus par une machine ;
- de complexité : certains problèmes ne peuvent être résolus en un temps raisonnable.

C'est par exemple sur ces points que reposent les technologies cryptographiques et le fameux problème "P=NP" à $1.000.000.

Programme
Logique des propositions et des prédicats : Formules, modèles, satisfiabilité, validité, prouvabilité
Exemples de théories: arithmétique et théorie des ensembles

Modèle de calcul : machines de Turing, calculabilité, décidabilité, théorème de l'arrêt, machine universelle
Classes de complexité : P, NP, réduction de problème, NP-complétude, problèmes SAT

Niveau requis
Pas de pré requis.
En revanche, ce cours est un pré-requis pour les thématiques "Algorithmique et optimisation" et "Langages, preuves, calcul" du PA Info en 3ème année.

Modalités d'évaluation
Examen sur table.

Responsable : Olivier BOURNEZ
Période : Automne
Nombre d'heures : 36
Crédits ECTS : 4
Dernière mise à jour : dimanche 15 avril 2012