

Diplôme d'Ingénieur
CNAM

Spécialité
INFORMATIQUE

Édition 2002

FILIÈRE INFORMATIQUE

Responsable de filière :

BERNARD LECUSSAN

Déroulement du cursus

<http://www.ipst.fr/formation/cnam/informatique/default.htm>

Le Cycle A est constitué de 7 unités de valeurs (plus une expérience professionnelle de deux ans) sanctionné par l'octroi d'un Diplôme de Premier Cycle Technique (DPCT) homologué au niveau III. Ce cycle A nécessite 2 à 3 ans d'études. Le niveau requis pour y accéder est le baccalauréat scientifique et si possible quelques connaissances en informatique.

Le Cycle Probatoire est constitué de 6 unités de valeurs et permet d'acquérir les connaissances scientifiques et techniques de base dans la spécialité. Ce cycle nécessite environ 3 ans d'études .

À l'issue des 4 premières unités de valeurs, 2 voies sont possibles :

- la préparation du Diplôme d'Etudes Supérieures techniques (DEST) homologué niveau II (bac + 4)
- la préparation du diplôme d'ingénieur CNAM homologué niveau I (Bac + 5)

Le DEST : avoir acquis les 6 unités de valeurs du cycle probatoire et une expérience professionnelle de 2 années

Le diplôme d'ingénieur - 2 étapes : l'examen probatoire et le mémoire d'ingénieur

L'examen probatoire :

- Avoir acquis les 4 premières unités de valeurs d'une des 3 dominantes et impérativement la 1/2 unité de valeur de communication. Une expérience professionnelle de 2 ans est exigée à la date de soutenance de l'oral probatoire.
- Passer l'examen probatoire. Il permet de déterminer si l'élève possède les qualités scientifiques, de synthèse, d'expression exigées d'un futur ingénieur. Pour cet examen probatoire, l'auditeur prépare en 6 semaines un rapport (sujet imposé par le responsable de la filière) qu'il soutient devant un Jury. En cas de succès, l'auditeur peut proposer un sujet de mémoire.

Le mémoire :

- Avoir effectué les deux demi - valeurs de cours et les deux demi - valeurs de management (du cycle probatoire)
- Avoir effectué **le Cycle d'Approfondissement** constitué de la valeur C, d'une demi - valeur de communication , d'un test d'anglais (le Bulat)
- **Réalisation d'un mémoire.** Lors de la soutenance, une expérience professionnelle de 3 ans dont 2 dans la spécialité, à un niveau suffisant est exigée.

CYCLE A

Admission et prérequis

Avoir le niveau d'un baccalauréat scientifique ou celui du cours Eléments de Mathématiques niveau 3 et avoir l'habitude du raisonnement mathématique.

Objectifs

Apprendre les notions de base sur les relations, l'algèbre de Boole et les fonctions booléennes, les dénombrements et les probabilités combinatoires, la récurrence, ainsi qu'un peu d'arithmétique.

Contenu

Généralités

Ensembles, éléments, parties d'un ensemble, fonctions, opérations sur les ensembles.

Dénombrements

Cardinal d'un ensemble, ensembles finis, ensembles dénombrables.

Arrangements, combinaisons, permutations, formule du binôme.

Probabilités combinatoires

Épreuves, événements, lois de probabilité, probabilités conditionnelles, indépendance, essais répétés.

Relations

Relation d'équivalence, relation d'ordre, diagramme de Hasse.

Treillis, algèbre de Boole, théorème de Stone.

Fonctions booléennes, forme canonique disjonctive, systèmes d'équations booléennes.

Chaînes de contacts, portes.

Simplification des formules, méthode de Karnaugh.

Logique

Propositions, connecteurs, formes propositionnelles.

Prédicats, quantificateurs.

Récurrences, définitions récursives.

Arithmétique

Division euclidienne, nombres premiers, PGCD, PPCM, identité de Bezout, fractions continues.

Admission et prérequis

Avoir l'habitude du raisonnement mathématique et avoir les connaissances de la demi-valeur Mathématiques pour l'informatique A1 (toutefois il n'est pas obligatoire d'avoir été reçu à Mathématiques pour l'informatique A1 pour s'inscrire).

Objectifs

Apprendre les automates finis, les codes détecteurs et les codes correcteurs, les graphes et surtout les arbres binaires.

Contenu

Matrices

Matrices à coefficients numériques, à coefficients binaires, à coefficients modulo 2.
Opérations sur ces matrices : somme et produit.

Automates finis

Caractères, chaînes de caractères, langages.

Opérations sur les langages : somme, produit, étoile.

Langages réguliers.

Automates, états, fonction de transition, langage de l'automate.

Automates finis non déterministes, automates finis non déterministes avec transition spontanée.

Détermination d'un automate.

Construction d'automates finis, théorème de Kleene, simplification d'un automate.

Codes détecteurs et codes correcteurs

Distance de Hamming, erreur de transmission, codage par blocs, correction et détection.

Codages linéaires, représentation matricielle, tableau standard, syndromes, codes cycliques.

Graphes

Graphes orientés, graphes non orientés, degrés chemins circuits, cycles, représentations matricielles.

Arbres, racine, arbres binaires, codes de Huffman.

Admission et prérequis

Avoir le niveau de la valeur de cours Mathématiques pour l'informatique A1 ou avoir le niveau d'un baccalauréat C, D, H, ou de l'attestation de réussite à Éléments de mathématiques niveau 3 et être inscrit à Mathématiques pour l'informatique A1. Ce cours étudie les concepts des langages informatiques et s'adresse aux personnes qui ont déjà une expérience de la programmation.

Objectifs

Ce cours a pour principal objectif de donner aux étudiants les concepts de base de la programmation moderne.

Pour cela, il s'appuie sur l'apprentissage d'un pseudo-langage afin d'acquérir une "démarche algorithmique" nécessaire pour le développement de petites et moyennes applications.

Enfin, tous les algorithmes sont codés dans le langage de programmation ADA.

Contenu

1. Introduction
2. Base du langage de description
3. Chaîne de fabrication d'un programme exécutable
4. Présentation du langage de programmation ADA
5. Les objets ou données
6. Les entrées / sorties
7. La programmation structurée
8. Manipulation des types d'objet
9. Adresse mémoire et pointeur de données
10. Les procédures et fonctions
11. Gestion des erreurs par exceptions
12. Algorithmes de recherche et de tris sur les tableaux
13. La récursivité
14. Les paquetages
15. La généricité
16. Les fichiers

Admission et prérequis

Avoir obtenu l'unité de valeur Mathématiques pour l'Informatique A1. Être inscrit parallèlement à l'unité de cours Algorithmique programmation A.

Objectifs

Acquérir la maîtrise de la mise en oeuvre des principaux concepts de base de la programmation moderne à partir de séances d'exercices sur machine, compléments essentiels pour la compréhension du cours Algorithmique programmation A.

Contenu

Moyens

Utilisation du langage ADA sur des machines UNIX. Le système n'est pas la priorité essentielle. Il sert de support initial pour l'accès aux compilateurs. Une séance d'initiation sert à présenter le système et les commandes principales (éditeur de texte, listage, création de répertoire...)

Organisation

Séances hebdomadaires d'1h30 en salle machines pendant toute l'année scolaire, avec encadrement. Des accès aux machines en libre service en dehors des séances sont vivement recommandées.

Structure

Plusieurs séries d'exercices sont regroupées dans un polycopié disponible en début d'année. Chaque élève travaille à son propre rythme. L'évaluation des connaissances est faite à travers des projets soutenus en fin de chaque semestre.

Programme

Présentation de l'environnement ADA.
Programme en une seule unité (type simple).
Fonctions, types enregistrement, tableau, généricité.
Procédures, passage de paramètres.
Modules, environnement. Réutilisabilité.

Admission et prérequis

Avoir le niveau des unités de valeur Algorithmique programmation A1 (cours et TP).

Objectifs

Donner les notions fondamentales de structures de données et de leur utilisation, et montrer comment les implanter à bon escient dans un langage de programmation de haut niveau. Faire comprendre l'importance de la spécification rigoureuse des structures de données, le pourquoi de l'étude de la complexité des algorithmes qui les manipulent, les principes de mise en oeuvre de ces structures.

Contenu

Notions préliminaires

Rappel des propriétés et caractéristiques essentielles des supports de mémorisation, tels que la mémoire centrale, les disques et les bandes. Notion de complexité des algorithmes: mesure d'efficacité en fonction de la taille du problème. Notion de type abstrait de données : syntaxe (signature), sémantique et implantation.

Les structures de données

Les structures séquentielles et les structures arborescentes. Définition sous forme de type abstrait, propriétés indépendantes de l'implantation. Principaux algorithmes liés à ces structures. Différentes techniques d'implantation de ces structures : avantages et inconvénients.

L'utilisation des structures

Principaux algorithmes de tri. Généralités et méthodes simples. Méthodes efficaces. Mesures et comparaisons entre ces algorithmes.

Principes de la recherche d'informations. Recherche séquentielle dans une liste quelconque. Recherche dichotomique dans une liste ordonnée pour laquelle on dispose de l'accès par le rang. Gestion d'un tas : solution efficace pour rechercher le plus petit élément d'un ensemble.

Utilisation de structures arborescentes pour la recherche. Les arbres binaires de recherche : recherche, adjonction et suppression. Évaluation de la complexité logarithmique en moyenne de ces opérations, et comparaison avec les structures séquentielles. Évaluation de la complexité au pire linéaire: amélioration par rééquilibrage donnant les arbres AVL. Analyse des opérations simples de rotation ponctuelles pour conserver l'équilibre.

Généralisation des arbres AVL aux arbres balancés pour prendre en compte une caractéristique des disques : la taille des blocs transférés. Application aux fichiers séquentiels indexés.

Recherche utilisant la notion de hachage : principes et méthodes de résolution des collisions.

Remarque : Implantations proposées au moyen de paquetages ADA génériques disponibles en machine (ou module C++), pour que les élèves puissent les utiliser lors de travaux pratiques personnels, et apprennent ainsi les notions fondamentales de réutilisation du logiciel.

Admission et prérequis

Avoir le niveau des unités de valeur Algorithmique programmation A (cours et TP).
Concerne des informaticiens d'application désirant acquérir une vision cohérente de l'architecture des systèmes.

Objectifs

Fournir des bases solides de compréhension des processus internes et externes et amener les élèves à une bonne utilisation de l'ensemble de l'outil informatique de base.

Contenu

Architecture des Machines

Les fondements logiques. Introduction à l'architecture des machines. La notion d'architecture externe et interne. Le codage des informations et des nombres. Les circuits logiques et les composants : circuits combinatoires (algèbre de Boole) et séquentiels (machines d'états finis), circuits cellulaires (mémoires et logiques programmables), circuits complexes (microprocesseurs). Éléments de technologie : circuits intégrés & VLSI.

L'architecture des processeurs. Le monoprocesseur : chemin des données, instruction, adressage, séquençement synchrone et asynchrone. La notion de micro-machine et de microprogrammation. L'évolution des processeurs RISC. Les systèmes d'entrées-sorties : entrées-sorties programmées, mode canal, processeurs d'entrée-sortie. Les contrôleurs de périphériques. Les systèmes d'interruption. Les principaux périphériques et la hiérarchie des mémoires.

L'architecture système. Les architectures multiprocesseurs : interconnexion entre processeurs, bus, multitraitement maître-esclave, multiprocesseurs symétriques, introduction au parallélisme synchrone et asynchrone . La gestion des hiérarchies de mémoires : concepts de cache et mémoires virtuelles, pagination et gestion déterministes. Introduction aux réseaux et architectures réparties : modèle OSI, concept de réseau local.

Architecture des systèmes d'exploitation

La chaîne de production de programme. Notion de traduction des langages de programmation. L'édition de liens : module transmutable, notion de lien, bibliothèque, notion de recouvrement, les références croisées. Le chargement. Les autres outils de la chaîne de production : aide à la mise au point, préprocesseurs et macrogénérateurs, le «make»...

Le programme et les objets externes. La notion de flot : ses différentes formes et l'établissement de la liaison avec l'objet externe. L'implantation des objets externes sur disque : séquentielle avec extensions, ou ensemble de blocs de taille fixe. La représentation de l'espace libre et la notion de quantum. La désignation des objets externes : notion de volume, notion de répertoire (désignation dans et sur le support). Arborescence de fichiers. Protection et sécurité des objets.

Le programme et son environnement physique. Notion de processus et de hiérarchie de processus. Notion de ressources et d'états d'un processus. Mécanismes de synchronisation des processus. L'interblocage et le problème de famine. Le partage de la mémoire centrale et la notion de multiprogrammation. Les limites du partitionnement de la mémoire. La segmentation et les mécanismes de pagination et leur utilisation par le système.

Admission et prérequis

Posséder les autres unités de valeur constituant le tronc commun des spécialités Informatique d'entreprise ou Génie informatique ou être inscrit aux unités manquantes, sauf Algorithmique programmation A1 cours et TP.

Objectifs

Analyser et réaliser un projet informatique de taille importante et de type professionnel avec les concepts du génie logiciel. Maîtriser un environnement et langage de programmation de haut niveau supportant les concepts «modernes» de programmation (mécanisme d'abstraction, composants logiciels...). Apprendre à communiquer entre les différentes équipes de projet dans le cadre de la réutilisation des composants logiciels développés par chaque équipe.

Contenu

Assistance algorithmique (A.A.)

Présentation du langage de programmation utilisé, cycle de vie de développement d'un projet informatique. Présentation des outils et des concepts d'analyse - Aide à l'analyse du projet.

Construction de programmes (conception, spécification, programmation), aide à l'analyse et à la spécification du projet (fonctionnelle, organique, découpage en modules, programmation de leur spécification et de leur implémentation), présentation et discussion des travaux par les équipes de projet (réutilisabilité des composants logiciels).

Travaux pratiques (UNIX)

Séminaires de présentation du système d'exploitation Linux (UNIX), utilisation du débogueur, de compilateurs et assistance à la réalisation du projet sur machine par les élèves.

Présentation des outils fonctionnant sur réseau INTERNET : messagerie électronique, ftp, Telnet, Usenet, Netscape.

Passage machine assisté

Programmation sur machines en libre service.

Admission et prérequis

Avoir le niveau des cinq unités de valeur du tronc commun des spécialités « Informatique d'entreprise » ou « Génie informatique » du cycle A d'informatique.

Objectifs

Apprendre comment modéliser des problèmes notamment d'optimisation, issus de l'informatique et de la recherche opérationnelle, comment les résoudre à l'aide d'un algorithme et d'une structure de données appropriée.

Contenu

Les problèmes combinatoires : généralités, difficultés.

Théorie des graphes et algorithmes de base

Introduction : vocabulaire et concepts de base (connexité, forte connexité, mise en ordre) ; Nombre cyclomatique, arbres et arborescences.

Représentations des graphes : matricielles (adjacence, incidence) ; listes (successeurs, prédécesseurs).

Les graphes en tant qu'outil de modélisation : exemples en informatique et en R.O.

Parcours des graphes : en largeur ; en profondeur ; applications ; détermination des composantes connexes, etc.

Fermeture transitive : détermination, méthode matricielle : algorithme de ROY-WARSHALL ; parcours en profondeur (cas d'un graphe sans circuit).

Initiation à la complexité dans le cas polynômial ; évaluation du nombre d'opérations.

Algorithmes d'optimisation dans les graphes valués

Chemins optimaux dans un graphe valué : algorithmes de FORD, de DIJKSTRA. Application : ordonnancements de projets (méthodes MPM et PERT).

Flots maximaux dans un réseau de transport : l'algorithme de FORD-FULKERSON (exemple, preuve, complexité).

Arbres couvrants de poids extrémal : algorithmes de KRUSKAL, de PRIM et de SOLLIN.

Programmes de transport (heuristiques et notion de « regret » ; algorithme du stepping-stone).

Problèmes d'affectation : la méthode hongroise (lien avec les flots maximaux).

Recherches arborescentes : en profondeur d'abord (Pb des reines sur l'échiquier) ; Branch and Bound : résolution du problème du voyageur de commerce (TSP) par l'algorithme de LITTLE et al.

Programmation linéaire

Définition, historique ; panorama des applications industrielles, performances et rentabilité. Approche géométrique ; caractérisation géométrique du cheminement vers le sommet optimum. Caractérisation algébrique d'un sommet. Méthode algébrique du simplexe ; méthode des tableaux (en se limitant au cas où le sommet 0 est admissible).

(Un approfondissement de ces concepts de base et des algorithmes associés fait l'objet d'UV des cycles probatoire puis d'approfondissement).

Admission et prérequis

Avoir le niveau des unités de valeur Algorithmique programmation A cours et TP et architecture des machines et systèmes informatiques A3.

Objectifs

Acquisition des connaissances de base en réseaux (architecture, principales normes).

Contenu

Base de la télécommunication, commutation, multiplexage.

Réseaux locaux.

Réseaux maillés et à grande distance.

Fonction de routage.

Administration réseau.

Internet : interconnexion réseaux.

Applicatifs réseaux et serveurs.

Admission et prérequis

S'adresse aux élèves en fin de cycle A souhaitant se spécialiser en Informatique d'entreprise.

Objectifs

Partie Bases de données

Savoir utiliser un SGBD relationnel et en particulier : définition d'un schéma relationnel et maîtrise du langage SQL.

Partie Processus d'informatisation

Donner une vue d'ensemble du processus d'informatisation, de la spécification à la maintenance, afin de permettre à l'auditeur de participer à un projet en connaissant l'ensemble des étapes.

Contenu

Bases de données

Conception et spécification d'un schéma relationnel

Modèle entité/association ; modèle relationnel ; création d'un schéma : la norme SQL2 ; normalisation.

Langages

Expression de requêtes : syntaxe et sémantique ; algèbre relationnelle ; le langage SQL.

Pratique d'un SGBD relationnel

L'environnement SQL2 (tables, vues, contraintes, utilisateurs, etc.) ; programmation avec SQL ; introduction à la concurrence d'accès.

Le cours est concrétisé par des travaux pratiques sur le SGBD ORACLE 8.

Processus d'informatisation

Introduction Définitions et concepts de base - notion de système - informations et données.

Systèmes d'information Rôle de l'informatique dans l'entreprise - typologie des systèmes d'information.

Processus d'informatisation

Les méthodes d'informatisation - les cycles de vie - les systèmes informatiques - les architectures-types - démarches d'informatisation.

Méthodes fondées sur les données : MERISE

Démarche MERISE - règles de gestion, d'organisation et techniques - modèles conceptuel, logique et physique des données et des traitements - études de cas.

Méthodes à objets : UML

Notion d'objet et de langages orientés objet (JAVA) - notion de cas d'utilisation - modèle de classe - diagrammes de séquence et de coopération.

Communication homme-machine Notion d'interface homme-machine - notion d'ergonomie - introduction à la construction des interfaces graphiques.

Conduite des projets (MCP).

CYCLE

PROBATOIRE

Admission et prérequis

Ce cours est destiné aux étudiants titulaires d'un diplôme de niveau BAC+2 dans une discipline scientifique autre que l'informatique et qui désirent suivre au CNAM une formation en vue d'obtenir un diplôme en informatique. Les titulaires d'un BAC+2 et de VARI seront dispensés de toutes les valeurs du cycle A d'informatique.

Objectifs

Ce cours présente de façon condensée les concepts de base de la discipline informatique enseignés dans plusieurs valeurs de premier cycle : principes de fonctionnement des ordinateurs, base de la programmation et de l'algorithmique. C'est donc un cours difficile qui exige d'une part des connaissances en mathématiques et d'autre part un investissement important. Les étudiants qui préféreraient un rythme moins soutenu peuvent remplacer la valeur VARI par des valeurs de cycle A. Le cours est illustré par des applications et des travaux pratiques.

Contenu

Architecture, système et réseaux

Introduction au fonctionnement des ordinateurs.

Systèmes d'exploitation : machine virtuelle, gestion des ressources.

Transmission numérique, exemples de protocoles.

Algorithmique et programmation

Conception d'algorithme et codage dans un langage de programmation.

Introduction aux principaux concepts des langages procéduraux : types, fonctions, procédures, exceptions, modules, généricité. Langage utilisé : C.

Algorithmique et structures des données

Conception d'algorithme, évaluation et complexité, automates, graphes.

Structures de données linéaires (listes, files, piles) et arborescentes (Tas, B-arbres) : construction, exploration, recherche.

Algorithmes de tri. Fichiers séquentiels indexés.

Admission et prérequis

Avoir le niveau Bac + 2 (DPCT du CNAM, DUT, BTS) en informatique.

Objectifs

Présenter des concepts, des méthodes et démarches indispensables pour de futurs ingénieurs chargés de conception et développement informatiques.

Contenu

- Graphes non valués

Concepts de base de la théorie des graphes.
Connexité, forte connexité, mise en ordre.
Fermeture transitive. Algorithme de ROY-WARSHALL.
Parcours des graphes (largeur, profondeur)
Exemples et applications.

- Optimisation dans les graphes valués

Chemins (algorithmes de FORD, DIJKSTRA, FLOYD).
Ordonnements (méthodes PERT et MPM).
Flot maximal. Flot à coût minimal.
Arbres optimaux

- Complexité des algorithmes : Introduction

- Introduction à la programmation linéaire

Définition, généralités. Reformulation de problèmes d'optimisation dans les graphes en P.L.
Algorithme du simplexe : fondements géométriques ; méthode algébrique.
Notion de dualité et application aux graphes.

- Eléments de théorie des files d'attente

Processus de naissance, loi de POISSON, loi exponentielle. Processus de naissance et de mort.
Files d'attente ; applications en informatique.

- Initiation aux réseaux de Petri

Au second semestre le cours MOCA B2 fait suite à cet enseignement.

Admission et prérequis

Avoir le niveau Bac + 2 (DPCT du CNAM, DUT, BTS) en informatique. Avoir le niveau de la demi-valeur MOCA B1.

Objectifs

Présenter des concepts, des méthodes, des démarches indispensables pour de futurs ingénieurs chargés de conception et développements informatiques.

Contenu

Méthodes de résolution de problèmes

Programmation dynamique.
Recherches arborescentes.
Heuristiques (gourmandes, tabou, recuit simulé).
Simulations.

Ordonnancements

Contraintes de ressources dans les ordonnancements - Ordonnancements d'atelier
- Applications en informatique.

Complexité des problèmes

Classes P, NP - Équivalence et réductions entre problèmes - Problèmes NP-complets, NP-difficiles - Théorème de COOK.

Réseaux de Petri

Définitions, exemples de modélisation, matrice associée, équation de mise à feu - Graphe des marquages accessibles, arborescence de KARP et MILLER. Semi-flots - Comportement d'un RdP (bornage, vivacité) - Modélisation et validation de systèmes informatiques distribués - GRAFCET.

Admission et prérequis

Avoir un niveau bac + 2 en informatique : DPCT, BTS, DUT...

Objectifs

Connaître dans le détail le fonctionnement d'un système informatique sous trois aspects : théorie du fonctionnement, transport dans l'espace et transport dans le temps. On appréciera ses performances et les apports de la normalisation.

Contenu

Histoire des idées et des évolutions techniques. Calculabilité, automate de Turing, apport de Von Neuman, machines séquentielles et machines parallèles.

Gestion comparée des hiérarchies de mémoires et des communications par deux modèles en sept couches : l'ordinateur comme machine logistique.
Mesures diverses et bancs d'essais.

Gestion des événements : systèmes d'interruptions, gestion des interruptions.

Normalisations.

Prévisions d'évolution à moyen et long terme.

Admission et prérequis

Avoir le niveau bac + 2 (DPCT du CNAM, BTS, DUT...) en informatique.

Objectifs

Donner un panorama complet des technologies et des méthodes permettant de réaliser des logiciels selon des critères de qualité définis à l'avance qui doivent permettre un traitement préventif des défauts.

Contenu

Les technologies du génie logiciel

Étude approfondie des différentes phases du cycle de vie du logiciel

Les méthodes du cycle de vie

Le développement : analyser, spécifier, concevoir, programmer, vérifier, intégrer, valider.

Modèle en cascade, modèle en spirale, modèle concurrent.

Les normes

Civiles : IEEE Software Engineering Standards Collection et ISO 9000. Militaires : GAM T17, DoD 2167A. Avionique : RTC/DO178. Sécurité : ITCSEC/ITSEC.

Automatisation du processus de développement : les outils

Activités répétitives et activités non répétitives. Gestion de configuration. Atelier de génie logiciel, maquettage et prototypage.

Cinématique et dynamique du processus de développement

Stabilité et instabilité du processus de développement. Rétro-ingénierie des logiciels. Intégration des logiciels.

Aspects socio-économiques du développement logiciel

Économie du génie logiciel

Données économiques quantitatives du développement de quelques grands systèmes. Modèles de coûts COCOMO. Facteurs influant sur les coûts. Critères qualité. Modèle qualitatif CQFD : coûts, qualité, fonctionnalités, délais. Gestion de projets.

Aspects stratégiques du développement des logiciels

Analyse du risque logiciel. Sûreté de fonctionnement. Qualité logicielle et contrôle qualité.

Facteurs organisationnels et humains

Management et organisation des projets logiciels.

Admission et prérequis

DPCT en Informatique ou équivalent (IUT ou BTS en informatique...)

Objectifs

Maîtriser l'utilisation d'un SGBD relationnel, et notamment l'utilisation du standard SQL. Comprendre tous les aspects de l'architecture d'un SGBD relationnel.

Contenu

Ce cours comporte trois parties:

1. Le modèle relationnel

- a) Rappels sur l'algèbre relationnelle.
- b) Les dépendances fonctionnelles et la conception d'un bon schéma.
- c) Étude approfondie du langage de requêtes standard SQL.
- d) Le calcul relationnel; application à SQL.

2. Les SGBD relationnels

- a) Architecture générale : les différentes fonctions, les différents niveaux, le stockage physique des données, les index, algorithmes de jointure.
- b) Évaluation et optimisation des requêtes (en prenant comme exemple un système du commerce)
- c) Concurrence d'accès et reprise sur pannes.

3. Compléments possibles

- a) Modèles relationnels étendus aux types abstraits et norme SQL2.
- b) Immersion de SQL dans un langage de programmation sur un exemple.
- c) Introduction à l'interopérabilité et à la répartition des données.
- d) Introduction au Tuning et à l'administration d'un SGBD.
- e) Introduction à la technologie orientée-objet.

Admission et prérequis

Avoir le niveau bac + 2 (DPCT du CNAM, BTS, DUT...) en informatique.

Objectifs

Le cours est conçu en deux parties complémentaires. La première partie traite les aspects fondamentaux des phases de spécification, prototypage, conception détaillée et validation des algorithmes. La deuxième partie traite de la conception générale, de l'architecture et de l'intégration des composants logiciels. Le langage Java est présenté et appliqué à l'implantation de problèmes spécifiés en première partie.

Contenu

1. La 1^{ère} partie (2 séances) est un rappel des principes du Génie Logiciel. Nous insistons tout particulièrement sur les cycles de vie et la place des méthodes de développement du logiciel dans les cycles de vie, et sur les qualités qu'elles permettent d'atteindre.
2. La 2^{ème} partie (6 séances) est constituée d'une présentation des concepts objets et de leur mise en oeuvre avec le langage JAVA, langage orienté-objet disponible sur tout système (Windows XX, Apple, Unix, Linux).
Attention : il ne s'agit pas d'apprendre à programmer en Java mais seulement de mettre en pratique les concepts objet en Java.
3. La 3^{ème} partie (6 séances) consiste en une présentation conjointe de la méthode orientée-objet Fusion, des 3 premières notations d'UML (Cas d'Utilisation, Diagrammes de séquences et Diagrammes de classes).
La présentation est incrémentale c'est à dire que l'on part d'exemples simples de conception en UML et on donne leur équivalent en Java. Puis on enrichit au fur et à mesure de la progression du cours les concepts UML et Java, en maintenant en permanence le lien entre conception et programmation grâce à Fusion.
4. La 4^{ème} partie (4 séances) constitue une application directe de la partie UML-Java. Elle consiste à traiter des exercices simples.
5. Dans la 5^{ème} partie (6 séances) sont approfondies et détaillées les notations UML. Les concepts nouveaux sont introduits et les liens avec la programmation Java sont explicités.
6. Enfin la dernière partie (6 séances) consiste à appliquer sur un exemple réaliste le processus de développement proposé par Fusion avec les notations UML.

Admission et prérequis

Avoir le niveau bac + 2 (DPCT du CNAM, BTS, DUT...) en informatique. Il est vivement conseillé d'avoir suivi au préalable la demi valeur Modélisation, optimisation, complexité et algorithmes B1 (10863) et la valeur Conception et développement du logiciel B3 (16472).

Objectifs

Permettre de comprendre les principaux éléments d'environnements comme ceux des systèmes UNIX et INTERNET. Obtenir des bases pour des compétences en : évolution des systèmes informatiques et des réseaux, ouverture et interfaçage des systèmes et réseaux, programmation d'applications comportant des processus coopérants, locaux ou distants (applications temps réel, applications réparties...), exploitation des systèmes, interconnexion des réseaux, l'administration des réseaux .

Contenu

Les réseaux et la répartition (exemple du réseau Internet)

Généralités sur les protocoles et architectures de réseau (exemple internet).

Les techniques de télécommunication physiques : problèmes de transmission sur un canal en présence de bruit, détection et correction des erreurs, produits ou normes (interfaces standards, modems).

Les réseaux locaux et le niveau liaison : accès au médium Ethernet, commutateurs de réseaux locaux, contrôle d'erreur, contrôle de flux, respect de l'ordre local.

Le niveau réseau et l'interconnexion de réseaux : adressage, routage, contrôle de congestion (exemple : IP, ATM).

Le niveau transport : gestion des connexions, contrôle d'erreur, contrôle de flux (exemple : TCP).

Systemes informatiques

Le système fournissant une machine virtuelle pour l'utilisateur. Processus et environnement d'exécution. Espace d'adressage : exemple d'Unix et Linux.

Le système fournissant des services communs et une plate-forme commune. Gestion des processus et partage des ressources. Structures de systèmes. Introduction et exemples sur l'allocation des ressources. Mesure de charge, d'occupation de l'espace d'adressage.

Ordonnancement des processeurs, (Exemples : Unix, Linus, Chorus, Temps réel). Ordonnancement des transferts disques.

Gestion de mémoire. Allocation. Placement par zone, par page. Mémoire virtuelle paginée.

Eroulement. Stratégies globales d'allocation de ressources. Localité, espace de travail.

Interblocage. Détection et prévention de l'interblocage. Prévention statique (classes ordonnées) et prévention dynamique (algorithme du banquier).

Paradigmes de la concurrence : exclusion mutuelle, producteurs-consommateurs, lecteurs-rédacteurs, dîner des philosophes. Système centralisé ou réparti.

Synchronisation par mémoire commune. Mécanismes élémentaires d'exclusion mutuelle. Sémaphore, moniteurs (aperçu), objets protégés de ADA95 (aperçu), méthodes synchronisées de JAVA.

Programmation de la synchronisation par «design patterns» avec des sémaphores (ou objets protégés ADA95). Propriétés, coopération entre processus, producteurs-consommateurs. Lecteurs-rédacteurs, dîner des philosophes. Les techniques de programmation. Sémaphores privés. Allocation de mémoire. Solutions en programmation par objets.

Synchronisation par messages. Cas d'UNIX et de LINUX. Boîte aux lettres. Client-serveur. Rendez-vous ADA. «Design patterns» pour les paradigmes.

Synchronisation par messages dans les systèmes répartis. Ordre partiel causal, ordre total à la Lamport, algorithme réparti d'exclusion mutuelle.

Admission et prérequis

Avoir le DPCT Informatique ou tout autre diplôme équivalent et terminer le cycle probatoire dans l'année.

Objectifs

1. Offrir une formation pratique, immédiatement utilisable, aux outils et langages les plus couramment utilisés dans le milieu informatique industriel. L'ensemble des outils de développement, disponibles sur une station de travail isolée ou connectée à un réseau, sont présentés.
2. Réaliser un projet informatique : de sa spécification à son implantation en passant par sa documentation et les phases de tests.

Contenu

Introduction aux méthodes du génie logiciel Acquisition de la maîtrise d'une station de travail connectée à un réseau (Unix, Internet) en vue du développement de logiciel (C++).

Le système Unix
Outils de l'internet
Introduction au langage C++
Programmation système
Communication inter-processus
Installation d'applications

Conception et réalisation d'un projet informatique sur l'un des thèmes : réseaux, systèmes distribués, recherche opérationnelle, compilation, multimédia.

Admission et prérequis

Avoir le niveau bac + 2 (DPCT du CNAM, BTS, DUT...) en informatique.

Objectifs

Présenter des notions de recherche opérationnelle et d'aide à la décision indispensables pour de futurs ingénieurs décideurs, responsables de projets.

Contenu

Graphes et ordonnancements en gestion de projets

Rappels des concepts élémentaires de théorie des graphes. Problème du chemin de valeur optimale entre deux sommets. Ordonnement de projets : méthodes PERT et MPM (chemin critique, marges). Traitement des contraintes cumulatives (budget).

Programmation linéaire et applications à l'entreprise

Généralités : origine, domaines d'application, pertinence.
Introduction géométrique puis algébrique à l'algorithme du simplexe.
Problème de la base initiale. Dualité. Analyse en sensibilité (paramétrages).

Analyse multicritère et systèmes interactifs d'aide à la décision (SIAD)

Méthodologies, concepts fondamentaux. Méthodes ELECTRE, «Goal-programming».
Présentation des SIAD (intérêts, limites).

Éléments de théorie des files d'attente et de sûreté de fonctionnement

Loi de Poisson, exponentielle. File d'attente M/M/1 et applications.
Fiabilité des composants, des systèmes (notions).
Paramètres de la sûreté de fonctionnement.

Admission et prérequis

Avoir le niveau bac + 2 (DPCT du CNAM, BTS, DUT...) en informatique.

Objectifs

Former des ingénieurs informaticiens à l'utilisation et au choix d'un SGBD. L'accent est mis sur l'utilisation des SGBD, la conception d'une base de données centralisée ou répartie, la structuration logique et physique d'une base de données relationnelle, l'administration des bases de données, l'optimisation des performances et l'évaluation du coût des opérations. Les défis modernes en matière d'architecture et d'utilisation des SGBD seront étudiés. Des illustrations seront faites par le biais des TP.

Contenu

Définitions et approches générales aux bases de données

Modélisation conceptuelle de données : Le modèle E-R et le modèle EE-R : Règles de construction du MCD, les mécanismes d'abstraction, La démarche de constitution d'un MCD

Introduction au modèle relationnel et passage du modèle E-R au modèle relationnel

Théorie de normalisation du modèle relationnel : Objectif, Conception d'un bon schéma relationnel, les dépendances fonctionnelles, les formes normales, l'algorithme par décomposition en 3^e forme normale, Plus loin que la 3^e forme normale

Mise en oeuvre des bases de données relationnelles : Les opérateurs de l'algèbre relationnelle et Initiation à SQL

Structures logique et physique d'une base de données relationnelle (oracle) : Structure d'une BD ORACLE, correspondance entre les structures logique et physique, Architecture fonctionnelle de SGBDR, les optimiseurs de requêtes, gestion des transactions, gestion des accès concurrents, verrouillage des données et gestion de reprise après incident

Approches à la gestion des bases de données réparties ou fédérées : les architectures des bases de données réparties, conception d'une base de données répartie, techniques utilisées dans la répartition des données

Administration et optimisation d'une BD relationnelle : niveaux d'administration, optimisation et amélioration des performances, méthodes d'accès, optimisation des requêtes SQL, choix des index et quelques astuces

Conclusion : Panorama des SGBD du marché, principaux Critères de choix d'un SGBDR, du SGBDR au SGBDOO

Admission et prérequis

Avoir le niveau bac + 2 (DPCT du CNAM, BTS, DUT...) en informatique.

Objectifs

Donner un panorama complet des technologies et des méthodes permettant de réaliser des logiciels selon des critères de qualité définis à l'avance qui doivent permettre un traitement préventif des défauts.

Contenu

Les technologies du génie logiciel

Étude approfondie des différentes phases du cycle de vie du logiciel

Les méthodes du cycle de vie

Le développement : analyser, spécifier, concevoir, programmer, vérifier, intégrer, valider.

Modèle en cascade, modèle en spirale, modèle concurrent.

Les normes

Civiles : IEEE Software Engineering Standards Collection et ISO 9000. Militaires : GAM T17, DoD 2167A. Avionique : RTC/DO178. Sécurité : ITCSEC/ITSEC.

Automatisation du processus de développement : les outils

Activités répétitives et activités non répétitives. Gestion de configuration. Atelier de génie logiciel, maquettage et prototypage.

Cinématique et dynamique du processus de développement

Stabilité et instabilité du processus de développement. Rétro-ingénierie des logiciels. Intégration des logiciels.

Aspects socio-économiques du développement logiciel

Économie du génie logiciel

Données économiques quantitatives du développement de quelques grands systèmes. Modèles de coûts COCOMO. Facteurs influant sur les coûts. Critères qualité. Modèle qualitatif CQFD : coûts, qualité, fonctionnalités, délais. Gestion de projets.

Aspects stratégiques du développement des logiciels

Analyse du risque logiciel. Sûreté de fonctionnement. Qualité logicielle et contrôle qualité.

Facteurs organisationnels et humains

Management et organisation des projets logiciels.

Admission et prérequis

Avoir le niveau bac + 2 (DPCT du CNAM, BTS, DUT...) en informatique.

Objectifs

Fournir les bases méthodologiques nécessaires à la conception et à la réalisation des systèmes d'information d'entreprise.

Contenu

Introduction aux méthodologies des systèmes d'information

- Introduction aux systèmes d'information
- Les méthodologies de conception de systèmes d'information
- Le cycle de vie d'un système d'information
- Les niveaux d'abstraction d'un système d'information.

La méthode MERISE

- La modélisation des données
- La modélisation des traitements
- Enchaînement des modèles
- Les outils AGL (Atelier de Génie Logiciel)

L'approche objet

- Les concepts de base
- Panorama des concepts objet
- Introduction à UML
- Les outils AGL

Admission et prérequis

Avoir le niveau bac + 2 (DPCT du CNAM, BTS, DUT...) en Informatique.

Objectifs

Acquérir les connaissances de base en matières de réseaux et de systèmes de communication. Comprendre la conception des réseaux en entreprise et maîtriser les principaux concepts de l'informatique communicante. Unité de valeur obligatoire de la dominante Ingénierie et Intégration Informatique (III).

Contenu

Première partie

Architectures et réseaux (OSI, SNA, TCP-IP).

Réseaux étendus, Réseaux locaux, transport sur les réseaux, Interconnexion de réseaux hétérogènes, Mise en oeuvre (performances, planification, administration et gestion).

Deuxième partie

La programmation répartie en mode message (exemple interface SOCKET, session OSI) et en appel de procédure distante (exemple DCE, CORBA).

Les transformations de formats (exemples de présentation, BER, IDL CORBA, CDR).

Protocoles de sécurité (KERBEROS). Techniques de cryptographie (DES, RSA).

Transfert de fichiers et systèmes de fichiers répartis : NFS, FTAM.

Transactionnel réparti : DTP, TUXEDO

Introduction aux applications internet

Messagerie et annuaires (exemples DNS, SMTP et X400, X500)

Web, protocole HTTP, langage HTML, XML.

Échange de données informatisées EDI.

Accès aux bases de données distantes (CLI, ODBC).

Administration de réseaux (SNMP).

Introduction aux collecticiels (groupware, workflow management).

Admission et prérequis

Avoir le niveau fin de cycle probatoire et avoir suivi les cours Informatique cycle probatoire B6 et B8.

Objectifs

Au travers des conférences, aborder des thèmes d'actualité différents et permettre aux élèves de manier des outils et de présenter un rapport et une conférence.

Contenu

Introduction aux méthodes du génie logiciel. Acquisition de la maîtrise d'une station de travail connectée au réseau (Unix, Internet) en vue du développement de logiciel (C++).

Le système Unix
Outils de l'internet
Introduction au langage C++
Programmation système
Communication inter-processus
Installation d'applications

À l'issue de ces conférences les élèves étudieront un thème, le mettront en oeuvre sous forme de travaux pratiques avec les outils logiciels.

Admission et prérequis

Avoir un niveau bac + 2 en informatique : DPCT, BTS, DUT...

Objectifs

Connaître dans le détail le fonctionnement d'un système informatique sous trois aspects : théorie du fonctionnement, transport dans l'espace et transport dans le temps. On appréciera ses performances et les apports de la normalisation.

Contenu

Histoire des idées et des évolutions techniques. Calculabilité, automate de Turing, apport de Von Neuman, machines séquentielles et machines parallèles.

Gestion comparée des hiérarchies de mémoires et des communications par deux modèles en sept couches : l'ordinateur comme machine logistique.

Mesures diverses et bancs d'essais.

Gestion des événements : systèmes d'interruptions, gestion des interruptions.

Normalisations.

Prévisions d'évolution à moyen et long terme.

Admission et prérequis

DPCT en Informatique ou équivalent (IUT ou BTS en informatique...)

Objectifs

Maîtriser l'utilisation d'un SGBD relationnel, et notamment l'utilisation du standard SQL. Comprendre tous les aspects de l'architecture d'un SGBD relationnel.

Contenu

Ce cours comporte trois parties:

1. Le modèle relationnel

- a) Rappels sur l'algèbre relationnelle.
- b) Les dépendances fonctionnelles et la conception d'un bon schéma.
- c) Étude approfondie du langage de requêtes standard SQL.
- d) Le calcul relationnel; application à SQL.

2. Les SGBD relationnels

- a) Architecture générale : les différentes fonctions, les différents niveaux, le stockage physique des données, les index, algorithmes de jointure.
- b) Évaluation et optimisation des requêtes (en prenant comme exemple un système du commerce)
- c) Concurrence d'accès et reprise sur pannes.

3. Compléments possibles

- a) Modèles relationnels étendus aux types abstraits et norme SQL2.
- b) Immersion de SQL dans un langage de programmation sur un exemple.
- c) Introduction à l'interopérabilité et à la répartition des données.
- d) Introduction au Tuning et à l'administration d'un SGBD.
- e) Introduction à la technologie orientée-objet.

Admission et prérequis

Avoir le niveau bac + 2 (DPCT du CNAM, BTS, DUT...) en informatique.

Objectifs

Donner un panorama complet des technologies et des méthodes permettant de réaliser des logiciels selon des critères de qualité définis à l'avance qui doivent permettre un traitement préventif des défauts.

Contenu

Les technologies du génie logiciel

Étude approfondie des différentes phases du cycle de vie du logiciel

Les méthodes du cycle de vie

Le développement : analyser, spécifier, concevoir, programmer, vérifier, intégrer, valider.

Modèle en cascade, modèle en spirale, modèle concurrent.

Les normes

Civiles : IEEE Software Engineering Standards Collection et ISO 9000. Militaires : GAM T17, DoD 2167A. Avionique : RTC/DO178. Sécurité : ITCSEC/ITSEC.

Automatisation du processus de développement : les outils

Activités répétitives et activités non répétitives. Gestion de configuration. Atelier de génie logiciel, maquettage et prototypage.

Cinématique et dynamique du processus de développement

Stabilité et instabilité du processus de développement. Rétro-ingénierie des logiciels. Intégration des logiciels.

Aspects socio-économiques du développement logiciel

Économie du génie logiciel

Données économiques quantitatives du développement de quelques grands systèmes. Modèles de coûts COCOMO. Facteurs influant sur les coûts. Critères qualité. Modèle qualitatif CQFD : coûts, qualité, fonctionnalités, délais. Gestion de projets.

Aspects stratégiques du développement des logiciels

Analyse du risque logiciel. Sûreté de fonctionnement. Qualité logicielle et contrôle qualité.

Facteurs organisationnels et humains

Management et organisation des projets logiciels.

Admission et prérequis

Avoir le niveau bac + 2 (DPCT du CNAM, BTS, DUT...) en informatique.

Objectifs

Approches qualitative et quantitative des systèmes d'exploitation et de communication. Conception et fonctionnement des systèmes d'exploitation centralisés et répartis, spécificités des systèmes temps réels. Introduction à la programmation système.

Ensemble cohérent de valeurs pour la spécification et la validation d'applications réparties : UV n° 19320, n° 19786, n° 16874 et n° 20001.

Contenu

Introduction générale

Structure des systèmes informatiques.
Structure des systèmes d'exploitation.

Gestion de processus

Processus : concepts, ordonnancement, opérations sur les processus.
Processus coopératifs, threads, communications inter-processus.

Ordonnancement de l'unité centrale

Concepts et critères d'ordonnancement.

Synchronisation de processus

Section critique, sémaphores, problèmes classiques, moniteurs.

Interblocage

Prévention, détection, correction...

Gestion de la mémoire

Swapping, allocation, pagination, segmentation.
Mémoire virtuelle.

Systèmes de fichiers

Interfaces des systèmes de fichiers et implémentation.

Protection et sécurité dans les systèmes

Systèmes distribués

Structure des réseaux et structure des systèmes répartis.
Système de fichiers répartis.

Admission et prérequis

Avoir le niveau bac + 2 (DPCT du CNAM, BTS, DUT...) en Informatique.

Objectifs

Acquérir les connaissances de base en matières de réseaux et de systèmes de communication. Comprendre la conception des réseaux en entreprise et maîtriser les principaux concepts de l'informatique communicante. Unité de valeur obligatoire de la dominante Ingénierie et Intégration Informatique (III).

Contenu

Première partie

Architectures et réseaux (OSI, SNA, TCP-IP).

Réseaux étendus, Réseaux locaux, transport sur les réseaux, Interconnexion de réseaux hétérogènes, Mise en oeuvre (performances, planification, administration et gestion).

Deuxième partie

La programmation répartie en mode message (exemple interface SOCKET, session OSI) et en appel de procédure distante (exemple DCE, CORBA).

Les transformations de formats (exemples de présentation, BER, IDL CORBA, CDR).

Protocoles de sécurité (KERBEROS). Techniques de cryptographie (DES, RSA).

Transfert de fichiers et systèmes de fichiers répartis : NFS, FTAM.

Transactionnel réparti : DTP, TUXEDO

Introduction aux applications internet

Messagerie et annuaires (exemples DNS, SMTP et X400, X500)

Web, protocole HTTP, langage HTML, XML.

Échange de données informatisées EDI.

Accès aux bases de données distantes (CLI, ODBC).

Administration de réseaux (SNMP).

Introduction aux collecticiels (groupware, workflow management).

Admission et prérequis

Avoir le niveau fin de cycle probatoire ou celui d'une maîtrise informatique.

Objectifs

Mise en application des connaissances acquises lors du cycle probatoire sous forme de conduite d'un projet par petits groupes. L'objectif principal est d'être capable de mener à bien une étude de sa conception à sa réalisation en utilisant des méthodes et outils de développement professionnels qui seront présentés. Les thèmes applicatifs seront proposés afin que cet enseignement puisse être aussi une ouverture de l'informatique vers d'autres spécialités.

Nouveau thème cette année : **systèmes embarqués**

Contenu

Conduite d'un projet sur architecture client/serveur

Compléments de cours :

- Le modèle de développement d'une application et le cycle de vie : recensement des besoins, la phase de conception, de programmation, des tests, la phase d'intégration et de qualification, l'installation, exploitation et maintenance.
- Les outils de conception : les moyens de représentation de l'architecture, des flots de données, des processus, d'enchaînement des tâches - La simulation et le prototypage - Les environnements de programmation - La production de documents.
- Systèmes d'exploitation multitâches, multi-utilisateurs : UNIX.
- Systèmes temps réel
- Les applications réparties : Le modèle client/serveur - La communication entre machines (les sockets) - RPC - Protocole TCP/ IP - Internet.
- Le langage C++

Projet :

Le projet suivra les étapes du cycle de vie d'un logiciel avec production des documents appropriés.

La mise en oeuvre de l'architecture client/serveur sous TCP/IP avec utilisation des sockets et la programmation en C sous UNIX seront communes à tous les projets.

Des projets systèmes embarqués seront proposés cette année.

Cours 12625 - COMMUNICATION CULTURE
(Préparation à l'examen probatoire) B1
0.5 valeur

Département : **Communication, culture, expression**
Chaire ou service : **Communication, culture, expression**

Admission et prérequis

Être sur le point de présenter l'examen probatoire et être agréé par l'enseignant.

Objectifs

Préparation à la rédaction d'un document de synthèse, entraînement à la soutenance orale. La formation commence avant la connaissance des sujets de façon à se terminer avant la date de soutenance orale. La formation fait l'objet d'une attestation de présence exigée pour le probatoire.

Contenu

Entraîner à la rédaction et à la soutenance du mémoire de probatoire.

Donner des outils pour améliorer l'écrit et l'oral.

Pratique de l'introduction et de la conclusion, du paragraphe, de la synthèse de documents etc.

Structuration et plan

Repères théoriques sur les deux types de textes utilisés : le texte informatif et le texte argumentatif.

Repères textuels et grammaticaux sur les difficultés de rédaction rencontrées.

Simulation de soutenance sous forme de mini-exposés de stagiaires (sur sujets libres) suivie de questions-réponses. Analyse des prestations.

Réflexion sur la situation de communication propre au probatoire.

Inscription à la scolarité.

IMPORTANT
cette 1/2 UV doit être
impérativement obtenue avant
de passer l'oral probatoire

Admission et prérequis

Demi-valeur obligatoire pour l'obtention du diplôme d'ingénieur CNAM. Elle doit être complétée par la demi-valeur Management social et humain pour l'entreprise B2. Inscription directe pour les élèves reçus à l'examen probatoire et priorité aux auditeurs engagés dans le cycle d'approfondissement du cursus conduisant au diplôme d'ingénieur.

Aucun prérequis en matière de techniques de management.

Objectifs

Sensibiliser les ingénieurs et cadres de formation initiale technique aux aspects économiques de leur métier. Plus précisément, les ouvrir à la complexité de l'environnement socio-économique de l'entreprise ; leur faire connaître les outils de gestion des activités : les aider à développer un comportement adapté.

Contenu

Management d'une activité : les bases du calcul économique

Introduction : les messages de l'analyse économique des marchés.

Des concepts pour décrire une activité économique et effectuer un diagnostic de ses résultats - L'activité du manager : contextes, théories et pratiques de la décision et de l'action.

Économie de la production et de l'investissement et politiques de croissance interne - Gestion prévisionnelle et pilotage de l'activité.

Management de l'entreprise face au marché

Dynamique des situations de concurrence et sources de l'action stratégique - Les modalités de l'action stratégique - L'innovation et le développement des produits - L'action commerciale - La gestion des relations industrielles (des achats à la coopération-concurrence).

Management des activités dans l'entreprise

De l'économie de l'organisation au management des équipes de travail et des unités opérationnelles - Le management des projets et des affaires - Le management par la qualité - Les démarches de conduite du changement.

La pédagogie alterne les apports de connaissance magistraux, les exercices d'application et l'étude de cas.

Admission et prérequis

Aucun prérequis en matière de techniques de management. Priorité aux auditeurs engagés dans le cycle d'approfondissement du cursus conduisant au diplôme d'ingénieur.

Objectifs

Sensibiliser les ingénieurs et cadres techniques aux aspects humains de leur métier. Initier à la complexité sociale et humaine de l'entreprise. Maîtriser des méthodes et des outils d'animation des équipes. Les aider à développer un comportement adapté.

Contenu

L'entreprise et les sciences sociales et humaines

L'entreprise et son organisation : structures, systèmes socio-techniques, fonctions - La fonction ressources humaines : politique sociale, stratégie, systèmes participatifs, économie de l'emploi - Analyse et évaluation du travail humain - Organisation et conditions de travail - L'ingénieur et le management humain.

Méthodes et outils du management social et humain

Analyse du travail humain : le cas des personnes handicapées - Hygiène et sécurité du travail - Les contraintes juridiques de la gestion des relations individuelles et collectives - Conduite d'une équipe et règlement des conflits - L'évaluation des compétences - Recueil et traitement des besoins de formation.

Éléments de droit du travail

Hygiène et sécurité

Admission et prérequis

Enseignement de base en intelligence artificielle, ce cours s'adresse aux étudiants n'ayant reçu aucune formation dans cette discipline et ayant le niveau de fin de cycle probatoire en informatique.

Objectifs

Ce cours présente les principes des grandes méthodes de l'intelligence artificielle et explique comment les appliquer pour résoudre des problèmes n'ayant pas de solution algorithmique (demi-valeur d'approfondissement de l'option modélisation RO-IA : optimisation en informatique, RO et programmation linéaire avancée ou reconnaissance des formes).

Contenu

Introduction

Définition et champ de l'intelligence artificielle.
Le niveau «connaissance» de Newell.
Notion d'agent intelligent.

Résolution de problèmes

Systèmes de productions, algorithme de filtrage : RETE.
Arbre de recherche, méthodes aveugles.
Démonstration automatique de théorèmes : résolution, unification, PROLOG.
Méthodes heuristiques : A*, AO*, Minmax, Alpha-béta, GBB.
Satisfaction et propagation de contraintes.
Analyse des buts et des moyens : GPS.

Représentation et utilisation des connaissances : systèmes experts

Classement des connaissances utilisées.
Représentation des connaissances.
Exemple de système expert (de diagnostic)
Conception de systèmes experts : introduction à la méthodologie KADS.

Apprentissage symbolique

À partir d'exemples.
Par analogie.

Admission et prérequis

Avoir suivi le cycle probatoire d'informatique ou niveau équivalent.

Objectifs

Approfondir des thèmes majeurs de la R.O. : La programmation linéaire et l'optimisation dans les graphes, tant au plan pratique que théorique.

Contenu

Présentation de problèmes industriels. Rappels des bases de la programmation linéaire (P.L.) et de l'algorithme du simplexe. Dualité, relations d'exclusion. Génération de colonnes. Initiation aux méthodes intérieures. Éléments de P.L. en nombres entiers.

Optimisation dans les graphes : chemins, flots, transports, affectation, tournée. Liens avec la P.L.

Réseaux de Petri et P.L.

Admission et prérequis

Avoir le niveau du cycle probatoire : Dominante Ingénierie et Intégration comportant les cours : Informatique cycle probatoire B6 et B8 et être agréé par l'enseignant. Les demi-valeurs Informatique systèmes d'information B1 et B3 constituent une préparation recommandée.

Objectifs

L'objectif principal de ce cours est de former des chefs de projet chargés de la définition de la stratégie, de la conception, de la mise en oeuvre et de l'évolution des systèmes d'information. À cette fin, les différentes méthodes appliquées à chaque phase du cycle de vie du système d'information seront étudiées. Tous les aspects de l'ingénierie des systèmes d'information seront étudiés. Tous les aspects fondamentaux de l'ingénierie des systèmes d'information, tant en ce qui concerne la dimension stratégique, managériale que technologique, seront analysés et illustrés. Le cours sera articulé autour des méthodes mises en oeuvre dans les trois phases principales d'un système d'information : le schéma directeur du système d'information, la conception et le développement des différentes applications composant le système d'information, la maintenance, l'évaluation, l'audit et l'évolution du système d'information.

Contenu

Schéma directeur des systèmes d'information

Introduction aux systèmes d'information

Élaboration des schémas directeurs des systèmes d'information : la méthode RACINES

Reconfiguration des processus et systèmes d'information : méthode de la pyramide

Conception et développement des applications

Méthodes et outils d'analyse des besoins

Méthodes et outils de spécifications fonctionnelles

Méthodes d'estimation des charges et des délais

Conception avancée des systèmes d'information

Modèles et méthodes de conception des systèmes d'information

Méthode de rétroconception des systèmes d'information

Démarche de classification de schémas

Méthode d'intégration des données et des schémas

Développement des systèmes d'information : la méthode RAD

Méthode de tests et de vérification formelle

Méthode d'intégration des composants d'un système d'information

Mise en oeuvre des systèmes d'information : techniques et outils

Maintenance, audit, évolution et évaluation

Les techniques de maintenance

L'audit des systèmes d'information

L'évolution des systèmes d'information : principes et méthodes.

Dans le cadre du cours, les auditeurs seront amenés à réaliser un projet qui donnera lieu à la rédaction d'un mémoire et à une présentation devant les enseignants et les étudiants.

Admission et prérequis

Sur agrément. Avoir des connaissances en systèmes d'exploitation, génie logiciel technique et architectures machines. Une connaissance de niveau élémentaire d'ADA est souhaitable.

Objectifs

Savoir développer des systèmes embarqués dans des produits industriels ou grand public (temps réel, téléphonie, carte à puce, assistant personnel, terminaux INTERNET...) en utilisant les technologies JAVA. Acquérir des compétences et savoir-faire dans l'ingénierie et la mise en oeuvre de tels systèmes.

Contenu

Introduction aux systèmes embarqués et au langage JAVA

La langage JAVA pour les systèmes embarqués.

Méthodologie de conception orientée Objet.

Catalogue de modèles réutilisables. Classification, description statique en UML. Les modèles utilisés par les librairie standard JAVA.

La machine virtuelle JAVA : principes et architecture

Jeu d'instructions, implémentation des objets, gestion de la mémoire et gestion de la concurrence.

Intégration d'une machine virtuelle développée en C sur un ordinateur de poche et sur une carte à puce.

Conception orientée objet et programmation de systèmes embarqués

Modèles de conception réutilisables. Conceptions d'interface graphique et de librairie spécifiques.

Système d'exploitation en JAVA et programmation concurrente

Les modèles de conception spécifiques.

Systèmes embarqués et temps réel : prise en compte de contraintes temporelles.

Développement de noyaux et intégration de librairies natives.

Intégration d'une machine virtuelle sur un exécutif Temps réel industriel.

Intégration de la machine JAVA dans un environnement distribué

Intégration sur Internet / intranet ou réseau spécialisé. Pilotage et contrôle distants de périphérique. Téléchargement de code et données à travers le réseau. «Plug and Play» d'une machine JAVA sur Internet. Les systèmes embarqués et JavaBeans.

Intégration logicielle et/ou matérielle : principes, partitionnement

Processeur JAVA

Technologies JAVA enseignées

JNI (JAVA Native Interface), RMI (Remote Method Invocation), JINI, EmbeddedServer, JavaBeans.

Contenu

Première partie Spécification

Modélisation des systèmes réactifs et des processus parallèles :

Modèle synchrone (Lustre, Esterelle), Modèle asynchrone : Réseau de Pétri, Systèmes de transitions

Expression des propriétés des systèmes réactifs : Sûreté, vivacité, logique temporelle

Spécification et vérification des propriétés :

Vérification par preuve et vérification par modèle

Vérification automatique par modèle, procédure de décision, méthode de tableau

Deuxième partie : sûreté de fonctionnement des systèmes réactifs

1. Définitions - terminologie - concepts de base
2. Sécurité et protection des informations
3. Fiabilité du logiciel : Présentation d'une démarche pour le suivi de la fiabilité du logiciel et l'optimisation de la politique de test
4. Analyses de risques
5. Modélisation des événements redoutés par la méthode des arbres de fautes
6. Quantification de la sûreté de fonctionnement par blocs diagrammes de fiabilité
7. Modélisation de systèmes

Admission et prérequis

Ouvert aux élèves-ingénieurs pour lesquels cette valeur est obligatoire

Objectifs

Améliorer ses compétences dans les domaines de l'expression et la communication orale et écrite, de la culture générale.

Le choix d'une option se fait avec l'aide d'un enseignant en fonction des besoins de l'étudiant jugés prioritaires (entretien d'orientation en septembre pour définir les modules du 1^{er} semestre).

La validation suppose une participation active de l'étudiant aux séances de formation.

Contenu

Les options proposées sont :

Communication dans la vie professionnelle et sociale

Réseaux de communication, stratégies, travail en groupe, conduite d'entretien et de réunions.

Communication écrite et vie professionnelle

Compte-rendus, rapports, rapports techniques.

Approche de la communication en entreprise à travers l'écrit.

Entraînement à écrire plus efficacement en situation professionnelle.

Recherche et projets

Aide pour la rédaction des mémoires et la soutenance orale devant un jury.

Maîtrise de l'écrit : normes, techniques, repères culturels

Parole et développement personnel

Communication et sociétés

Les grandes théories contemporaines, compréhension des mécanismes à partir de mise en situations et d'exercices.

Modules généraux de 40h au 1^{er} semestre.

Modules complémentaires (20 ou 40 heures au 2nd semestre)

Le BULATS

Business Language Testing Service

Niveau II exigé

Évaluation du niveau linguistique des auditeurs

Le test comprend les exercices suivants :

ÉCOUTE :

Comprendre de courts extraits
Prendre des messages téléphoniques
Comprendre un enregistrement plus long

LECTURE :

Comprendre des avis et de courts extraits de lettres, formulaires, etc.
Trouver une information dans des textes brefs
Comprendre un texte plus long

GRAMMAIRE ET VOCABULAIRE

Remplir les blancs dans un texte (vocabulaire)
Remplir les blancs dans un texte de grammaire
Corriger les erreurs dans un texte

Niveau 2

Maîtrise limitée mais effective de la langue dans des situations familières, exemple ; peut prendre part à une réunion sur des sujets familiers, surtout s'il s'agit d'un simple échange d'informations.