

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

**MINISTERE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE**

OFFRE DE FORMATION MASTER

ACADEMIQUE

Etablissement	Faculté / Institut	Département
Université Hassiba Benbouali - Chlef	Faculté Des Sciences Exactes et Informatique	Informatique

Domaine : M. I

Filière : Informatique

Spécialité : Ingénierie du Logiciel (IL)

الجمهورية الجزائرية الديمقراطية الشعبية
وزارة التعليم العالي و البحث العلمي

عرض تكوين ماستر

ماستر أكاديمي

القسم	الكلية/ المعهد	المؤسسة
الإعلام الألي	كلية العلوم الدقيقة والإعلام الالي	جامعة حسيبة بن بو علي بالشلف

الميدان :رياضيات- إعلام ألي

الشعبة الإعلام الألي

التخصص : هندسة البرامج

Fiche d'organisation semestrielle des enseignements

1- Semestre 1 :

Unité d'Enseignement	VHS	V.H hebdomadaire				Coeff	Crédits	Mode d'évaluation	
	14-16 sem	C	TD	TP	Travail Personnel			Continu (1/3)	Examen (2/3)
UE fondamentales									
UEF1 (P) : ING 1 1		4h30	1h30	3h	3h		10		
Algorithmique avancé et complexité	90h	3h	1h30	1h30	1h30	3	6	x	x
Génie logiciel et gestion de projets	45h	1h30		1h30	1h30	2	4	x	x
UEF2 (P) : ING 1 2		3h	1h30	3h	3h		10		
Systèmes d'exploitation	67h30	1h30	1h30	1h30	1h30	3	6	x	x
Systèmes d'informations avancés	45h	1h30		1h30	1h30	3	4	x	x
UE Méthodologie									
UEM1 (P) : ING 1 3		3h	3h	1h30	2h		9		
Modélisation et évaluation des performances des systèmes	45h	1h30	1h30		1h	2	4	x	x
Architecture et administration des bases de données	67h30	1h30	1h30	1h30	1h	3	5	x	x
UE transversales									
UET1 (P) : ING 1 4		1h30			1h	1	1		
Anglais	22h30	1h30			1h	1	1	x	x
Total Semestre 1	382h30	12h	6h	7h30	9h		30		

2- Semestre 2 :

Unité d'Enseignement	VHS	V.H hebdomadaire				Coeff	Crédits	Mode d'évaluation	
	14-16 sem	C	TD	TP	Travail Personnel			Continu (1/3)	Examen (2/3)
UE fondamentales									
UEF1 (P) : ING 2 1		3h	3h	1h30	3h		10		
Systèmes et algorithmes répartis	67h30	1h30	1h30	1h30	1h30	3	6	x	x
Bases de données avancées	45h	1h30	1h30		1h30	2	4	x	x
UEF2 (P) : ING 2 2		3h	1h30	1h30	2h30		8		
Fondements de l'intelligence artificielle	45h	1h30	1h30		1h	2	4	x	x
Compilation	45h	1h30		1h30	1h30	2	4	x	x
UE méthodologie									
UEM1 (P) : ING 2 3		4h30	1h30	1h30	3h		11		
Sécurité informatique	67h30	1h30	1h30	1h30	1h	3	6	x	x
Preuve de programmes	22h30	1h30			1h	1	2	x	x
Validation et tests de logiciels	22h30	1h30			1h	1	3	x	x
UE transversales									
UET1 (P) : ING 2 4		1h30			1h		1		
Anglais	22h30	1h30			1h	1	1	x	x
Total Semestre 2	382h30	12h	6h	4h30	9h30		30		

3- Semestre 3 :

Unité d'Enseignement	VHS	V.H hebdomadaire				Coeff	Crédits	Mode d'évaluation	
	14-16 sem	C	TD	TP	Travail Personnel			Continu (1/3)	Examen (2/3)
UE fondamentales									
UEF1 (P) : ING 3 1		3h	3h	3h	3h		10		
Méthodes de conception et construction des programmes	67h30	1h30	1h30	1h30	1h30	3	5	x	x
Ingénierie des modèles	67h30	1h30	1h30	1h30	1h30	3	5	x	x
UEF2 (P) : ING 3 2		3h	1h30	3h	3h		09		
Systèmes Décisionnels et Entrepôts de données	67h30	1h30	1h30	1h30	1h30	3	5	x	x
Datamining	45h	1h30		1h30	1h30	2	4	x	x
UE méthodologie									
UEM1(P) : ING 3 3		4h30		4h30	3h		9		
Vérification formelle des systèmes	45h	1h30		1h30	1h	2	3		
Interface Homme-machine	45h	1h30		1h30	1h	2	3		
Module à option au choix (Représentation des connaissances, Ingénierie des connaissances, Systèmes d'informations, coopératifs, Traitement du langage naturel, Bases de données distribuées, Interface homme-machine,	45h	1h30		1h30	1h	2	3	x	x
UE transversales									
UET1 (P) : ING 3 4		3h			2h		2		
Ethique et déontologie de travail	22h30	1h30			1h	1	1		
Techniques d'expression et de communication	22h30	1h30			1h	1	1	x	x
Total Semestre 3	427h30	9h	4h30	6h	17h		30		

5- Récapitulatif global de la formation : (indiquer le VH global séparé en cours, TD, pour les 04 semestres d'enseignement, pour les différents types d'UE)

VH \ UE	UEF	UEM	UED	UET	Total
Cours	292H 30	180		90	462
TD	180	67H30			231
TP	225	112H30			252
Travail personnel	490	215		60	803
Autre (Stage)		290			290
Autre (séminaire)			21		21
Total	1187H30	865		150	
Crédits	87	29		4	120
% en crédits pour chaque UE	72,5%	24.16%		3.34%	100%

Programme détaillé par matière

Intitulé du master : Ingénierie du logiciel

Intitulé de la matière : Algorithmique avancé et complexité **Code : ALGO**

Semestre : S1 **Unité d'Enseignement : UEF1** **Code : ING11**

Nombre d'heures d'enseignement (90h sur 15 semaines)

Cours : 3 h/ Semaine TD : 1h30/semaine TP : 1h30/Semaine

Nombre d'heures de travail personnel pour l'étudiant : 1h30 (22h30 sur 15 semaines)

Nombre de crédits : 6 crédits

Coefficient de la Matière : 3

Objectifs de l'enseignement :

Ce module permet d'acquérir les notions nécessaires pour :

- *analyser et classer les problèmes de différents domaines*
- *construire la ou les solutions*
- *évaluer les différentes solutions en terme de calcul de complexité*
- *Choisir la meilleure solution*

Ces notions seront vues à travers l'étude de problèmes pris dans différents domaines de l'informatique tels que : Les réseaux, les bases de données, Algorithmique du texte ...etc.

Connaissances préalables recommandées :

L'étudiant doit connaître les bases de programmation et des algorithmes itératifs et récursifs ainsi que les structures de données fondamentales : tableaux, fichiers, piles, files, listes et arbres.

Contenu de la matière :

1. analyse des algorithmes

1.1 Introduction à l'algorithmique.

- 1.2 Calcul d'ordre des algorithmes et notations asymptotiques
- 1.3 Estimation de l'ordre du temps de calcul.

2. Algorithmes récursifs

- 1. Récursivité
 - 1.1 Définition
 - 1.2 Types de Récursivité
 - 1.3 Principe et dangers de la récursivité
 - 1.4 Non décidabilité de la terminaison
 - 1.5 Importance de l'ordre des appels récursifs
- 2. Dérécursivation
 - 2.1 Récursivité terminale
 - 2.2 Récursivité non terminale
- 3. Diviser pour régner
 - 3.1 Principe
 - 3.2 Résolution des récurrences

3. Arbres et graphes

- 3.1 Arbres binaires
- 3.2 Arbres AVL
- 3.3 Arbres B
- 3.4 Arbres Red Black
- 3.5 Graphes

5. Recherche d'une sous-chaîne de caractère dans un texte

- 5.1 Algorithme naïf.
- 5.2 Algorithme Rabin-Karp.
- 5.3 Algorithme Boyer-Moore.
- 5.4 Algorithme Knuth-Morris-Pratt.

Ce cours présente les concepts de base de l'algorithmique et de l'analyse des algorithmes. Les principales techniques de conception d'algorithmes seront aussi couvertes.

À la fin de ce cours, l'étudiant (e) aura assimilé les notions suivantes :

- techniques d'analyse de performance des algorithmes ;
- techniques de base de conception d'algorithmes : récursivité, , etc. ;
- algorithmes de recherche et d'extraction de l'information

Et sera capable de :

- analyser la performance des algorithmes pour faire un choix réfléchi entre divers algorithmes;
- concevoir des algorithmes simples en utilisant les techniques classiques de conception;
- analyser les résultats et faire des améliorations algorithmiques si possible et nécessaire.

Mode d'évaluation : Examens de moyenne durée, TP, exposés.

Références (*Livres et photocopiés, sites internet, etc*).

1. Introduction to algorithms. T. Cormen, C. Leiserson, R. Rivest. MIT Press, 2nd edition 2000.
2. Algorithms and theory of computation handbook, edited by M. Atallah, CRC Press, Purdue University, 1999.
3. Analysis of algorithms: an active learning approach. J.J.McConnell . Jones and Barlett Publishers, 2001.
4. Computational complexity. C.H. Papadimitriou, Addison Wesley, 1994.

Intitulé du master : Ingénierie du logiciel

Intitulé de la matière : Génie logiciel et gestion de projets Code : *GLGP*

Semestre : S1 Unité d'Enseignement : **UEF1** Code : **ING11**

Nombre d'heures d'enseignement (45h sur 15 semaines)

Cours : 1h30/ Semaine TD : - TP : 1h30/Semaine

Nombre d'heures de travail personnel pour l'étudiant : 1h30 (22h30 sur 15 semaines)

Nombre de crédits : 4 crédits

Coefficient de la Matière : 2

Objectifs de l'enseignement

- Permettre à l'étudiant de comprendre l'enjeu majeur de la gestion de projet.
- Initier l'étudiant au processus d'organisation et de planification.
- Entraîner l'étudiant à l'application de processus, méthodes et outils de planification.
- Initier l'étudiant aux environnements de gestion projet.

Connaissances préalables recommandées

Les techniques de développement de logiciels, à savoir :

- les notions de base du génie logiciel.
- Les techniques et les produits d'analyse et de conception du logiciel.
- Les techniques de programmation.

Contenu de la matière :

Notions de projet et gestion de projet.

Les modèles de gestion de projet

- Les modèles de basés sur les livrables.
- Les modèles basés sur le risque.

Les éléments de Gestion de projet

- Les enjeux de gestion de projet.
- Les activités de gestion de projet.
- La structure de gestion de projet.
- Les risques et la gestion de projet.

L'organisation des équipes de programmation

- L'organisation de base.
- Les outils de support.

Les éléments de la planification

- La productivité du programmeur.
- Echéance et jalonnement d'un projet.

Le processus de planification

- Découpage et coordination des activités.
- Les outils de planification (ordonnancement des activités et affectation des ressources).
- Les environnements de planification (ex : MSPROJECT).

Estimation des charges, délais et coût

- Les options alternatives : méthodes.
- La précision de la taille des programmes.
- Modèle d'estimation algorithmique.

Mode d'évaluation : contrôle continue et un examen écrit

Références (*Livres et photocopiés, sites internet, etc*).

Livres :

- Principles of software engineering management by Tom GILB Edition Lavoisier.
 - Software Engineering: A Practitioner's Approach by [Roger S Pressman](#).
 - [Software Project Management in Practice](#) by Pankaj Jalote.
 - [Génie logiciel: principes, méthodes et techniques](#) by [Alfred Strohmeier](#) et [Didier Buchs](#).
 - *Les environnements de gestion* : Microsoft Project, Planner, Gant project.
- Les sites* : - Software engineering Institute, SEI.org
- Gant.sourceforge.net

Intitulé du master : Ingénierie du logiciel

Intitulé de la matière : Systèmes d'exploitation Code : SE

Semestre : S1 Unité d'Enseignement : UEF2 Code : ING12

Nombre d'heures d'enseignement (67 H30 sur 15 semaines)

Cours : 1h30/ Semaine TD : 1h30/semaine TP : 1h30/Semaine

Nombre d'heures de travail personnel pour l'étudiant : 1h30 (22h30 sur 15 semaines)

Nombre de crédits : 6 crédits

Coefficient de la Matière : 3

Objectifs de l'enseignement

- Approfondir les différents concepts utiles pour la compréhension des différents composants d'un système d'exploitation et leurs interactions.
- Introduire l'étudiant aux problèmes de conception et de programmation d'un système d'exploitation.

Recommandations :

- Il est conseillé d'utiliser un système d'exploitation ouvert (UNIX par exemple) comme outil d'expérimentation des concepts étudiés.
- Prévoir des TPs pour la mise en application des concepts étudiés.

Connaissances préalables recommandées

Notions de base d'un système d'exploitation.

Contenu de la matière :

Chapitre 1: SYSTEMES DE FICHIERS

- + Rappels sur l'interface des systèmes de fichiers
- + Structure d'un système de fichiers (organisation, montage)
- + Organisation physiques des fichiers (allocation contiguë, chaînée, indexée)
- + Gestion de l'espace libre (vecteur binaire, liste chaînée, groupement)
- + Implémentation des répertoires (linéaire, table de hachage)
- + Gestion des fichiers actifs: partages de fichiers
- + Protection
- + SGF sous Unix

Chapitre 2: PROTECTION ET SECURITE

- + Protection
 - . Domaine de protection
 - . Matrices de droits
 - . Protection et langages évolués.
 - . Exemple de systèmes de protections
- + Sécurité
 - . Authentification
 - . Menaces
 - . Surveillance des menaces
 - . Cryptage

Chapitre 3 : SYNCHRONISATION DES PROCESSUS

- + Problème de l'exclusion mutuelle
- + Synchronisation
- . Sémaphores,
- . Evénements,
- . Moniteurs
- . Régions critiques
 - + Exemples sous UNIX

Chapitre 4 : COMMUNICATION ENTRE PROCESSUS

- + Partage de variables (modèle de producteur/ consommateur, lecteurs/rédacteurs)
 - + Boite aux lettres
 - + Echange de messages (modèle du client/ serveur)
 - + Communication sous Unix (partage de segments, tubes, files de messages, sockets)

Chapitre 5 : Elément d'administration d'un système d'exploitation

- + Installation (serveur et postes)
- + Configuration
- + Partage de ressources
- + Attribution des privilèges
- + Virtualisation

Mode d'évaluation : Examen écrit, travaux personnels notés.

Références (*Livres et polycopiés, sites internet, etc*).

- [1] M. J. Bach, traduit par G.Fellah, "Conception du Système UNIX," Masson et Prentice Hall, 1990.
- [2] J. Beauquier, B. Berard "Systèmes d'exploitation : Concepts et algorithmes" McGraw Hill, 1990.
- [3] Crocus, "Systèmes d'exploitation des ordinateurs," Dunod informatique 1975.
- [4] N. B. Fontaine, P. Hammes, "UNIX Système V: Système et environnement, Masson 1989.
- [5] S. Krakowiak, "Principes des systèmes d'exploitation des ordinateurs," Dunod informatique 1987.
- [6] J-L.Peterson, F.Silbershartz "Operating Systems Concepts," Addison-Wesly Publishing Company, Inc, 1983.
- [7] A. Silberschatz, P. B. Galvin "Principes des systèmes d'exploitation," 4 e Edition, Addison Wesley, 1994.
- [8] A. S. Tanenbaum, "Modern Operating Systems, " Second Edition Prentice Hall.

Intitulé du master : Ingénierie du logiciel

Intitulé de la matière : Systèmes d'information avancés **Code : SIA**

Semestre : S1 **Unité d'Enseignement : UEF2** **Code : ING12**

Nombre d'heures d'enseignement (45h sur 15 semaines)

Cours : 1h30/ Semaine TD : - TP : 1h30/Semaine

Nombre d'heures de travail personnel pour l'étudiant : 1h30 (22h30 sur 15 semaines)

Nombre de crédits : 4 crédits

Coefficient de la matière : 2

Objectifs du cours

- Comprendre ce qu'est un système d'information dans ses différentes dimensions
- Atteindre un bon niveau en programmation objet JAVA et utiliser des patrons de conception
- Utiliser naturellement les principaux diagrammes UML
- Savoir exprimer les besoins d'un système d'information avec des cas d'utilisation
- Connaître les principes généraux de toute méthode de conception orientée objet (processus unifié)

Pré-requis

Programmation orientée-objet, bases de données.

Contenu de la matière :

1. Rappels sur l'objet
2. Rappels sur la conception de SI
3. Unified Modeling Language (UML)
 - 3.1 Introduction à UML
 - 3.2 Diagrammes statiques
 - 3.3 Diagrammes dynamiques
 - 3.4 Concepts avancés.
4. Patrons de conception
5. Cas d'utilisation et expressions des besoins

- 6. Méthodes et processus de conception de S.I OO
 - 6.1 Processus de conception de S.I
 - 6.2 Processus unifiés
 - 6.3 Méthodes Agile

Modes d'évaluation

- Examen écrit final
- Contrôle continu : 2 travaux pratiques et 1 projet à rendre, contrôles rapides éventuels en cours d'année

Références

- Grady Booch, Ivar Jacobson, James Rumbaugh (2004) *UML 2.0 Guide de référence*, Campus Press, 774 p.
- Henri Chelli (2003) *Urbaniser l'entreprise et son Système d'information : Guide des entreprises agile*, Vuibert, 202 p.
- Alistair Cockburn (2002) *Agile Software Development*, Addison Wesley, 280 p.
- Eric Gamma, Richard Helm, Ralph Johnson, John Vlissides (1996), *Design patterns, Elements of Reusable Object-Oriented Software*, Addison-Wesley, 395 p.
- Ivar Jacobson, Grady Booch, James Rumbaugh (2000) *Le processus unifié de développement logiciel*, Eyrolles, 488 p.
- Craig Larman (2005) *UML 2 et les Design Patterns (3e édition)*, Pearson Education, 655 p.
- Chantal Morley, Jean Hugues, Bernard Leblanc, Olivier Hugues (2005) *Processus Métiers et systèmes d'information : Evaluation, modélisation, mise en oeuvre*, Dunod, 245 p.
- Pierre-Alain Muller, Nathalie Gaertner (2003) *Modélisation Objet avec UML (2ème édition)*, Eyrolles, 514 p.
- Robert Reix (2004) *Systèmes d'information et Management des organisations (5e édition)*, Vuibert, 487 p.
- Pascal Roques, Franck Vallée (2004) *UML2 en action (3ème édition)*, Eyrolles, 386 p.

Intitulé du master : Ingénierie du logiciel

Intitulé de la matière : Modélisation et évaluation des performances des systèmes

Code : MEPS

Semestre : S1

Unité d'Enseignement : UEM1

Code : ING13

Nombre d'heures d'enseignement (45 H sur 15 semaines)

Cours : 1h30/ Semaine TD : 1h30/semaine TP : -

Nombre d'heures de travail personnel pour l'étudiant : 1h (15 H sur 15 semaines)

Nombre de crédits : 4 crédits

Coefficient de la matière : 2

Objectifs du cours

L'objectif de ce module est de sensibiliser les étudiants aux problèmes de modélisation et d'évaluation des performances des systèmes réels tels les systèmes informatiques, les réseaux de communication et les systèmes de production. Il se propose de répondre aux questions suivantes : Pourquoi évaluer les performances d'un système ? Dans quels cas cela est-il nécessaire ? Comment modéliser un système ? Quel type de modèle utiliser ? Comment analyser le modèle ?.

Connaissances préalables recommandées :

Des notions de probabilités et statistiques.

Contenu de la matière :

- I. Problématique de l'évaluation de performances.
- II. Chaînes de Markov.
- III. Files d'attente.
- IV. Les réseaux de Petri simples (Analyse structurelle, comportementale, invariants).
- V. Les réseaux de Petri stochastiques, Analyse des performances.

Mode d'évaluation : Examen Ecrit, Contrôle continu.

Références :

B. Baynat, Théorie des files d'attente, Hermes 2000

G. Vidal-Naquet, A. Choquet-Geniet, Réseaux de Petri et Systèmes Parallèles, Armon Colin 1992
A. Choquet-Geniet, Les Réseau de Petri, un outil de modelisation Dunod 2006

Intitulé du master : Ingénierie du logiciel

Intitulé de la matière : Architecture et Administration de Bases de Données

Code : AABD

Semestre : S1 Unité d'Enseignement : UEM1 Code : ING13

Nombre d'heures d'enseignement (67h30 sur 15 semaines)

Cours : 1h30/ Semaine TD : 1h30/semaine TP : 1h30 /Semaine

Nombre d'heures de travail personnel pour l'étudiant : 1h (15 H sur 15 semaines)

Nombre de crédits : 5 crédits

Coefficient de la matière : 3

Objectifs du cours

Donner aux étudiants les notions pour d'une part concevoir et mettre en place des bases de données (BDD) et d'une part concevoir et réaliser des systèmes de gestion de base de données (SGBD)

Connaissances préalables recommandées

Les premières notions sur les bases de données : conception, création, et utilisation d'une base de données

Contenu de la matière :

I.RAPPELS :

1. Bases de données
 - 1.1 Définition
 - 1.2. Niveaux de description
2. Systèmes de gestion de bases de données
 - 2.1 Définition
 - 2.2 Fonctions
 - 2.3 Architecture générale

II. ARCHITECTURE DES S.G.B.DE DE PREMIERE GENERATION

1. Architecture des S.G.B.D de type hiérarchique
2. Architecture des S.G.B.D de type réseau CODASYL

III. ARCHITECTURE ET FONCTIONS DES SYSTEMES RELATIONNEL

1. Architecture des SGBD relationnels
2. Création des relations de base
3. Chargement des données
4. Définition des chemins d'accès
5. Dérivation des nouvelles relations
6. Catalogue de base
7. Stockage et accès aux données
8. Notions de transactions

IV. MEMOIRE RELATIONNELLE

1. Un modèle de mémoire relationnelle
2. Les méthodes d'accès
Séquentiel (rappel), Séquentiel indexe (ISAM), ARBRES (VSAM),
Hachage virtuel.
3. Implémentation des chemins d'accès dans les SGBD relationnels
4. Implémentation des opérateurs de l'algèbre relationnelle

V. OPTIMISATION DES REQUETES

1. Définition du problème
2. Rappels sur les propriétés des opérateurs algébriques
3. Techniques d'optimisation pour les langages algébriques
4. Technique de sélection optimale des chemins d'accès
5. Technique de décomposition des requêtes

VI. LE CONTROLE DES ACCEES CONCURRENTS

1. Définition du problème
2. Caractérisation des exécutions sans conflit
3. Algorithmes basée sur l'ordonnancement initial des transactions
4. Algorithmes de verrouillage

VII. RESISTANCE LES PANNES ET LA SECURITE DES DONNEES

1. Définition du problème
2. Types de pannes
3. Les procédures de reprise

4.La sécurité des données

Mode d'évaluation : Examen écrit, travaux personnels notés.

BIBLIOGRAPHIE :

1. G. Gardarin. *Bases de données. Les systèmes et leurs langages.* (Eyrolles, 1988).
2. J. Date. *Date on Database : Writings 2000-2006.* (Apress, 2007).
3. C. Delobel, M. Adiba. *Bases de données et systèmes relationnels.* (Dunod informatique, 1982).
4. S. Miranda et J.M. Busta. *L'art des bases de données. Tome1 et 2. Les bases de données relationnelles.* (Eyrolles, 1986).

Intitulé du master : Ingénierie du logiciel

Intitulé de la matière : Systèmes et Algorithmes répartis Code : SAR

Semestre : S2 Unité d'Enseignement : UEF1 Code : ING21

Nombre d'heures d'enseignement (67 h 30 sur 15 semaines)

Cours : 1h30/ Semaine TD : 1h30/semaine TP : 1h30/Semaine

Nombre d'heures de travail personnel pour l'étudiant : 1h30 (22h30 sur 15 semaines)

Nombre de crédits : 6 crédits

Coefficient de la matière : 3

Objectifs du cours

- Introduire les architectures parallèles et l'algorithmique distribuée.
- Présenter les concepts des systèmes d'exploitation distribués et leurs spécificités (allocation de ressources, communication, synchronisation, élection)

Recommandations :

- Prévoir des TPs pour la mise en application ou la simulation des concepts étudiés.

Connaissances préalables recommandées

Notions de base d'un système d'exploitation.

Contenu de la matière :

Chapitre 1: INTRODUCTION AUX ARCHITECTURES PARALLELES

- + Introduction
- + fondements (besoins des applications et limites des architectures simples)
- + Avantages et domaines d'application des architectures parallèles.
- + Classification des architectures parallèles (Flynn)

Chapitre 2: INTRODUCTION AUX SYSTEMES D'EXPLOITATION DISTRIBUES

- + Introduction, fondements
- + Services des systèmes d'exploitation distribués.

- + Présentation d'un système d'exploitation distribué (exemple : Chorus)
- + Concepts de base : ordre d'évènements, causalité, temps dans un système distribué, état d'un système distribué.

Chapitre 3: PROBLEMES FONDAMENTAUX DES SYSTEMES D'EXPLOITATION DISTRIBUES

- + Algorithmes de synchronisation dans un système distribué (Lamport, Ricart/Agrawala, Kawazaki).
- + Algorithmes d'allocation de ressources et d'interblocage un système distribué.
- + Algorithmes de terminaison de processus.
- +Algorithme d'élection de processus dans un système distribué.

Mode d'évaluation : Examen écrit, travaux personnels notés.

Références (*Livres et photocopiés, sites internet, etc*).

- [1] Ajay D. Kshemkalyani, "Distributed Computing, principles, algorithms and systems" Cambridge, 2008.
- [2] G. Coulouris "Distributed systems , Concepts and Design," .
- [2] S. Hariri, Manish Parashar "Tools and Environments for parallel and Distributed Computing" Wiley, 2003.
- [3] J-L.Peterson, F.Silbershartz "Operating Systems Concepts," Addison-Wesly Publishing Company, Inc, 1983.
- [4] A. Silberschatz, P. B. Galvin " Principes des systèmes d'exploitation," 4 e Edition, Addison Wesley, 1994.
- [5] A. S. Tanenbaum, " Modern Operating Systems, " Second Edition Prentice Hall.

Intitulé du master : Ingénierie du logiciel

Intitulé de la matière : Bases de données avancées Code : BDA

Semestre : S2 Unité d'Enseignement : UEF1 Code : ING21

Nombre d'heures d'enseignement (45h sur 15 semaines)

Cours : 1h30/ Semaine TD : 1h30/semaine TP :

Nombre d'heures de travail personnel pour l'étudiant : 1h30 (22h30 sur 15 semaines)

Nombre de crédits : 4 crédits

Coefficient de la matière : 2

Objectifs du cours

Ce cours a l'objectif ambitieux de permettre la découverte des différents aspects liés aux nouvelles tendances dans les bases de données :

- connaissances sur les bases de données et SGBD orientés objet
- connaissances sur les bases de données distribuées

Connaissances préalables recommandées

Connaissances sur les bases de données classiques : modélisation relationnelle, notion de transaction, langages de requête etc..

Contenu de la matière :

Chapitre I. INTRODUCTION

I. Faiblesse des SGBD relationnelles

II. les nouvelles perspectives des bases de données

Chapitre II. Base de données objet-relationnelles.

I. Introduction.

I. Les SGBDR.

I.1. Les avantages du modèle relationnel

I.2. Les faiblesses du modèle relationnel

II. Les SGBDOO

II.1. Les avantages des SGBDOO

II.2. Les inconvénients des SGBDOO

- II.3.Le passage entre SGBDR/SGBDRO/SGBDOO
- III. Les SGBDRO
 - III.1.Modèles NF2
 - III.2.Types utilisateurs
 - III.3.Les tables imbriquées dans le modèle relationnel-objet
 - III.4. Les tables objets
- IV. Le passage conceptuel vers relationnel-objet
 - IV.1.Classe.
 - IV.2. Attributs et méthodes
 - IV.3.Association 1 : N et 1 :1
 - IV.4.Association N : M.
 - IV.5. Composition.
 - IV.6.Héritage.
- V. Le modèle Objet relationnel et le SQL3.
 - V.1.Caractéristiques d'une table objet-relationnel
 - V.2.Création de type objet
 - V.3.Interrogation dans une table objet relationnelle.
 - V.4. Définition de types utilisant des références (ou pointeurs)
 - V.5.Fonction Deref et VALUE.
 - V.6.Simulation de l'héritage.
 - V.7.Tables imbriquées.
 - V.8.Les Tableaux (ou VARRAY)
- Conclusion du chapitre II.

Chapitre III. Les données semi-structurées.

Introduction.

I. Le langage XML.

I.1.Définition d'une DTD.

I.2. Validation d'un document XML.

II. Bases de données et XML.

II.1.Utilisation d'une base de données relationnelle.

II.2.Utilisation des bases de données intégrant des types XML.

II.3. Mapping Relationnel/XML.

III. Le passage entre le Relationnel et XML.

IV. Bases de données XML natives.

Chapitre IV. Les bases de données réparties.

I. Introduction.

II. SGBD réparti

II.1.12 règles de Date.

II.2.Fragmentation verticale et Fragmentation horizontale.

III. Migration vers une BDR..

III.1.Décomposition physique en BD locales.

III.2.Intégration logique des BD locales existantes.

III.3.Modèles d'intégration de bases de données.

Chapitre V. La gestion des accès concurrents

I.Le concept de transaction

II.Les états d'une transaction

III.Exécutions concurrentes

IV. Sériabilité

V. Définition des transactions dans SQL

Mode d'évaluation : Examen écrit, travaux personnels notés.

BIBLIOGRAPHIE :

1. C. Delobel, C. Lécluse, P. Richard *bases de données des systèmes relationnelles aux systèmes objets intereditions 1991*
2. Omran A. Bukhres, Ahmed K. Elmagarmid : *Object Oriented Multidatabase Systems: A solution for advanced applications* Prentice Hall 1996
3. Thomas Connolly, Carolyn Begg: *Systèmes de bases de données : approche pratique de la conception, de l'implémentation et de l'administration*, Eyrolles 2005

Kumar, Interscience mobile database Systems, Wiley 2006

Intitulé du master : Ingénierie du logiciel

Intitulé de la matière : Sécurité informatique Code : S/

Semestre : S2 Unité d'Enseignement : UEM1 Code : ING23

Nombre d'heures d'enseignement (67h30 sur 15 semaines)

Cours : 1h30/ Semaine TD : 1h30/semaine TP : 1h30/Semaine

Nombre d'heures de travail personnel pour l'étudiant : 1h (15 H sur 15 semaines)

Nombre de crédits : 6 crédits

Coefficient de la matière : 3

DESCRIPTION

Ce cours vise à introduire les concepts de base

- 1 - Concepts de base (les objectifs et la politique de la SI, Le mise en oeuvre et les mecanisme et Audit) et principe CIA
- 2 - Confidentialité : Cryptographie (chiffrement de donnees et de communication)
- 2 - Integrité : Hashage et Authenticité
- 4 - Avaibility (Disponibilité) : Continuite de service et plan apres catastrophe
- 5 - Applications (Réseaux, Protection des donnees, Authentification,..)

Objectifs du cours

La mise en place du commerce électronique requiert une infrastructure technologique (ordinateurs, équipement de réseau, systèmes d'exploitation et serveurs) basée sur les technologies Internet. L'objectif de ce cours est de présenter les concepts reliés à la mise en place d'une telle infrastructure technologique dans une entreprise de taille moyenne.

A la fin du cours, l'étudiant sera en mesure :

- de comprendre les concepts liés au commerce électronique;
- de comprendre la technologie associée aux réseaux et les technologies Internet en particulier;
- de comprendre la structure et le fonctionnement du réseau Internet;
- de maîtriser les problèmes et les défis liés à la sécurité des réseaux et les solutions existantes;
- d'analyser les derniers développements dans le domaine.

Chapitre I 1. INTRODUCTION A LA SEURITE INFORMATIQUE

1.1 Concepts

1.2 Architecture OSI de SI

1.3 Attaques / Services / Mecanismes

1.4 Modèle sécurité des données et des communications

Chapitre II CRYPTOGRAPHIE SYMMETRIQUE

2.1 Techniques Classiques

2.1.1 Cryptographie Classique

2.1.1 Rappel mathématique : arithmétique de Z_n

2.1.2 Techniques de la Substitution

2.1.3 Techniques de la Transposition

2.2 Cryptage par Bloc (BC)

2.2.1 Principes du BC

2.2.2 Data Encryption Standard (DES) Algorithme

2.1.1 Rappel mathématique : arithmétique des corps finis

2.2.3 Advanced Encryption Standard (AES) Algorithme

2.2.4 Modes Opérateurs du BC

2.3 Cryptage par Flot (SC) et Génération Pseudo Aléatoire des Nombres (PNG)

2.3.1 PNG et Générateurs PN

2.3.2 LFSR : Registre lineaire à decalage retroactif

2.3.2 RC4 Algorithme

Chapitre III CRYPTOGRAPHIE ASYMMETRIQUE

3.1 Cryptographie à clé publique (PKC) et RSA

3.1.1 Principes des Crypto systèmes PKC

2.1.1 Mathématique de SI: factorisation des nombres premiers

3.1.2 Algorithme RSA

3.1.3 Protocole Diffie-Hellman échange des clés

Chapitre IV INTEGRITE DES DONNEES

4.1 Fonctions de hachage cryptographique

4.2 Message Authentication Codes (MAC)

4.3 Signature électronique (DS)

4.4 Gestion et Distribution des clés

4.5 Protocoles d'authentification des utilisateurs

Chapitre V SECURITE SYSTEMES et RESEAUX

5.1 Attaques : Intrusions, Logiciels malveillants

5.2 Protection : Firewalls, Politique Sauvegarde, IDS,...

5.3 Sécurité de la couche applicative (PKI, PGP, HTTPS, ..)

5.4 Sécurité de la couche Transport (WEB, SSL, SSH,..)

5.5 Sécurité de la couche réseaux : IPsec et WIFI

Mode d'évaluation : Examen écrit, travaux personnels notés.

BIBLIOGRAPHIE :

1. RAYPORT, Jeffrey F. & JAWORSKI, Bernard J. – Introduction au commerce électronique – McGraw-Hill, 2000.
2. KIENAN, Brenda – E-commerce, Stratégies et solutions – Microsoft Press, 2001.
3. CHERKAOUI, Omar – La téléinformatique, – Chenelière McGraw Hill, 2ème édition.
4. SHERIF, M.H, SERHOUCHNI, A. – La monnaie électronique – Eyrolles.

5. PUGOLLE, G. – Les Réseaux – 2e édition, 2003, Eyrolles, juillet 2002 - 1088 pages.
6. KUROSE, James et ROSS, Keith – Analyse structurée des réseaux – Pearson Education, 2003.
7. COMER, D.E – Computer Networks and Internets – Prentice Hall. – <http://www.netbook.cs.purdue.edu/index.htm>
8. TANENBAUM, Andrew S. – Computer Networks – Prentice Hall, 1996 (en français chez Dunod, 1997).
9. SHARMA, Vivek , SHARMA, Rajiv – Developing e-Commerce Sites: An Integrated Approach – <http://www.awprofessional.com/>
10. DAVIS & BENAMATI – E-Commerce Basics: Technology Foundations and E-Business Applications – Addison-Wesley, Boston, MA, 2003.

Intitulé du master : Ingénierie du logiciel

Intitulé de la matière : Fondements de l'intelligence artificielle Code : FIA

Semestre : S2 Unité d'Enseignement : UEF2 Code : ING22

Nombre d'heures d'enseignement (45 H sur 15 semaines)

Cours : 1h30/ Semaine TD : 1h30/semaine TP : -

Nombre d'heures de travail personnel pour l'étudiant : 1h (15h sur 15 semaines)

Nombre de crédits : 4 crédits

Coefficient de la matière : 2

Objectifs de l'enseignement

L'étudiant après avoir acquis cette matière est capable de :

- Résoudre des problèmes d'Intelligence artificielle
- Concevoir des systèmes d'intelligence artificielle (systèmes experts, etc...)
- Pouvoir étudier les techniques d'intelligence artificielle avancées.

Connaissances préalables recommandées : Notions générales de l'informatique

Contenu de la matière

Chapitre1 : Introduction générale

- Définition de l'IA
- Histoire de l'IA
- Quelques applications de l'IA
- Etat de l'art de l'IA

Chapitre 2 : Calcul du 1^{er} ordre

- Définitions, Syntaxe et Sémantique
- Définition de Conséquence logique
- Substitution et unification
- Règle d'inférence
- Notion de Clause

- Résolution
- Quelques exemples d'utilisation du calcul de 1^{er} ordre
- Limites du calcul du 1^{er} ordre
- Introduction aux logiques non classiques

Chapitre 3 : Les systèmes de règles de production (SP)

- Définition d'un SP
- Procédure de base et contrôle dans les SP
- SP spécialisés (commutatifs/décomposables)
- Exemples de problèmes résolus par les SP

Chapitre 4 : Système de réfutation par résolution

- Position du problème
- Système de production pour les réfutations par résolution
- Stratégies de contrôle pour les méthodes de résolution (qcq stratégies)
- Stratégies de simplification (qcq stratégies)
- Exemples de problèmes résolus par ce système

Chapitre 5 : Stratégies de recherche

- Stratégie de retour arrière chronologique
- Stratégie de recherche avec graphe
- Procédure aveugle de recherche
- Procédure heuristique (informée) de recherche
- Utilisation des fonctions d'évaluation
- Algorithme A et A*

Chapitre 6 : Les systèmes experts

- Définition d'un système expert
- Méthodologie de construction de systèmes experts
- Quelques exemples de systèmes experts

Chapitre 7 : Planification en robotique

- Position du problème en robotique
- Description de l'état du monde, du but et des actions (format STRIPS)

- Résolution du problème en chaînage avant
- Table triangulaire
- Résolution du problème en chaînage arrière
- Notion de régression
- Buts interférants et buts décomposables
- Le système STRIPS
- Utilisation des systèmes de déduction

Chapitre 8 : Quelques langages d'IA

- Prolog
- Lisp

Chapitre 9: Problèmes CSP

- Définition de CSP
- Méthode de résolution des CSP

Chapitre 10 : Introduction à l'IA avancée.

Ce chapitre fait une petite introduction aux domaines d'IA avancée qui feront l'objet d'un cours en M2

Mode d'évaluation : Examen écrit, travaux personnels notés.

Références (*Livres et photocopiés, sites internet, etc*).

- *Principles of Artificial Intelligence* par J. Nilson
- *Essentials of Artificial Intelligence* par Morgan Kaufmann,
- *Artificial Intelligence : A new synthesis* par Morgan Kaufmann,
- *Artificial Intelligence: A Modern Approach* par Stuart Russell et Peter Norvig
- *aima.cs.berkeley.edu*

Intitulé du master : Ingénierie du logiciel

Intitulé de la matière : Compilation

Code : COMPIL

Semestre : S2

Unité d'Enseignement : UEF2

Code : ING22

Nombre d'heures d'enseignement (45h sur 15 semaines)

Cours : 1h30/ Semaine TD : - TP : 1h30/Semaine

Nombre d'heures de travail personnel pour l'étudiant : 1h30 (22h30 sur 15 semaines)

Nombre de crédits : 4 crédits

Coefficient de la matière : 2

Objectifs de l'enseignement

Le cours de compilation de Master 1ère année permet :

- *d'approfondir les notions supposées étudiées en Licence, tant en ce qui concerne l'analyse lexicale, syntaxique, que le typage.*
- *de se concentrer sur les parties propres d'un compilateur. Les transformations de programmes d'optimisation, leurs conditions d'application et l'analyse statique des programmes y sont également étudiées*

Connaissances préalables recommandées :

Théorie des langages, notion de compilation, Langage de programmation Pascal, C, C++

Contenu de la matière

- 1 - Rappels sur l'analyse lexicale.
- 2 - Rappels sur l'analyse syntaxique.
- 3 - Analyse sémantique, portée, tables des symboles.
- 4 - Allocation - Substitution
- 5 - Environnement d'exécution.
- 6 - Optimisation de code
- 5 - Génération de code.

Mode d'évaluation : Examen écrit, travaux personnels notés.

BIBLIOGRAPHIE

1. Aho A. Ullman J.D., *Principes des Compilateurs*, 1986 Edison

2. Andrew w. Appel, *Modern Compiler Implementation in ML*, Cambridge University Press 1998
 3. H. Drias *compilation cours et exercices* OPU 1992
 4. J. Levine, T. Masson, D. Brown, *lex&yacc*, Editions O'Reilly International Thomson 1995.
-

Intitulé du master : Ingénierie du logiciel

Intitulé de la matière : Preuve de programmes

Code : PVP

Semestre : S2 **Unité d'Enseignement : UEM1**

Code : ING23

Nombre d'heures d'enseignement (22h30 sur 15 semaines)

Cours : 1h30/ Semaine TD : - TP : -

Nombre d'heures de travail personnel pour l'étudiant : 1h (15 H sur 15 semaines)

Nombre de crédits : 2 crédits

Coefficient de la matière : 1

Objectifs de l'enseignement

L'objectif de ce cours est de familiariser l'étudiant à un raisonnement formel sur les programmes (en termes de correction et de complexité) et de lui enseigner une démarche rigoureuse de construction de programmes.

Connaissances préalables recommandées :

Des notions de mathématiques et de logiques

Contenu de la matière :

1. Introduction à la preuve de programme
2. Sémantique Opérationnelle
3. Correction partielle et Logique de Hoare
4. Terminaison de programme
5. Outils d'assistant à la preuve

Mode d'évaluation : Examen Ecrit, Contrôle continu

Références :

- 1. F. PAGAN « Formal specification of programming languages » printice-hall International, 1981.
- 2. E. HOROWITZ « Fundamentals of programming Languages » Computer Science Press, 1984.

Intitulé du master : Ingénierie du logiciel

Intitulé de la matière : Validation et tests de logiciels **Code : VTL**

Semestre : S2 **Unité d'Enseignement : UEM1** **Code : ING23**

Nombre d'heures d'enseignement (22h30 sur 15 semaines)

Cours : 1h30/ Semaine TD : - TP : -

Nombre d'heures de travail personnel pour l'étudiant : 1h (15h sur 15 semaines)

Nombre de crédits : 3 crédits

Coefficient de la matière : 1

Objectifs de l'enseignement

Ce cours est destiné à définir l'action de tester un logiciel et de présenter les différents tests de logiciels ainsi que la gestion et la planification de cette opération de test.

Connaissances préalables recommandées :

Aucunes.

Contenu de la matière

- 1 - Introduction
- 2- Les test fonctionnels
- 3- Les tests structurels dynamiques
- 4- Les outils pour l'automatisation du test
- 5- Gestion et planification des tests

Mode d'évaluation : Examen écrit final

BIBLIOGRAPHIE

- The Art of Software Testing. Glenford J Myers (Tom Badgett, Corey Sandler, Todd M. Thomas). John Wiley and Sons. 2004.
- Software Testing : A Craftsman's Approach. Paul C Jorgensen. CRC Press. 2002

- Le test des logiciels. Spyros Xanthakis, Pascal Régner, Constantin Karapoulos. Hermès. 2000.

Intitulé du master : Ingénierie du logiciel

Intitulé de la matière : Méthodes de conception et construction des programmes

Code : MCCP

Semestre : S3

Unité d'Enseignement : UEF1

Code : ING31

Nombre d'heures d'enseignement (67 H30 sur 15 semaines)

Cours : 1h30/ Semaine TD : 1h30/semaine TP : 1h30/Semaine

Nombre d'heures de travail personnel pour l'étudiant : 1h30 (22h30 sur 15 semaines)

Nombre de crédits : 5 crédits

Coefficient de la matière : 3

Objectifs de l'enseignement

Le programme couvre deux aspects :

- Conception d'algorithmes :

On présente les techniques à mettre en oeuvre pour trouver la solution d'un problème donné

- Construction de programmes :

On présente dans cette partie les différents modes de programmation qui existent, c'est à dire on s'intéresse à la forme des programmes. Pour chaque Forme, nous essayerons également de donner les preuves et les sémantiques.

Connaissances préalables recommandées :

Langages de programmation : PASCAL, C, LISP, PROLOG, SMALLTOCK

Contenu de la matière

I. Méthodes de conception de programmes

1. Concepts préliminaires

* O-notation

* Graphes et Arbres

2. Diviser pour régner

3. Récursion

4. Programmation dynamique

5. Backtracking

- * Breadth First Search
- * Depth First Search
- 6. Heuristiques
- * Best First Search
- * Branch and Bound Search
- * Optimal Search A*

II. Construction de programmes

1. Introduction
2. Programmation structurelle
3. Programmation fonctionnelle
4. Programmation logique
5. Programmation orienté objet
6. Méthodes et outils de spécification
7. Langage de spécification

Mode d'évaluation : Examen écrit final

BIBLIOGRAPHIE

1. A. Aho, J. Hopcroft, J. Ullman. "Structures de données et algorithmes". Interédition 1989.
2. Brassard, Ph Bizard. "Algorithmes : Analyse et Conception". Masson 1989.
3. P. Berlioux, Ph Bizard. "Algorithmique : Construction, Preuve et Evaluation des programmes" Dunod 1983.
4. Peyton jones. "Mise en oeuvre des langages fonctionnels de programmation". Masson 1990.

Intitulé du master : Ingénierie du logiciel

Intitulé de la matière : Ingénierie des modèles

Code : IM

Semestre : S3

Unité d'Enseignement : UEF1

Code : ING31

Nombre d'heures d'enseignement (67h30 sur 15 semaines)

Cours : 1h30/ Semaine TD : 1h30/semaine TP : 1h30/Semaine

Nombre d'heures de travail personnel pour l'étudiant : 1h30 (22h30 sur 15 semaines)

Nombre de crédits : 5 crédits

Coefficient de la matière : 3

RESUME :

L'IDM (Ingénierie dirigée par les modèles) est le domaine de l'informatique mettant à disposition des outils, concepts et langages pour créer et transformer des modèles.

Ce que propose l'approche de l'ingénierie des modèles (IDM, ou MDE en anglais pour Model Driven Engineering) est simplement de mécaniser le processus que les ingénieurs expérimentés suivent à la main. L'intérêt pour l'IDM a été fortement amplifié à la fin du XX-esiècle lorsque l'organisme de standardisation OMG (Object Modeling Group) a rendu publique son initiative MDA (Model Driven Architecture).

Ce module aborde l'IDM à la fois dans ces aspects modélisation, transformation et applications industrielles. Cet enseignement est composé de cours et TP, et s'appuie sur le développement d'un microprojet qui mettra en application les différents éléments présentés dans cet enseignement.

Objectifs de l'enseignement

Acquisition des connaissances élémentaires en IDM, étude d'outils de support à l'IDM et première expérience en développement par modélisation.

Connaissances préalables recommandées :

UML

Contenu de la matière

1. Introduction à l'IDM, présentation des micro-projets
2. Introduction à l'IDM
3. Méta-modélisation

4. Principes des Transformations en IDM
5. Principes des Transformations : ATL, QVT
6. IDM dans l'entreprise (exemples et applications)

BIBLIOGRAPHIE :

- Jean-Marie Favre, Jacky Establier, Mireille Blay-Fornarino, editors. "L'ingénierie dirigée par les modèles : au-delà du MDA" 236 pages, Hermes-Lavoisier, Cachan, France, feb 2006 ISBN 2-7462-1213-7

Intitulé du master : Ingénierie du logiciel

Intitulé de la matière : Systèmes Décisionnels et Entrepôts de données

Code : SDED

Semestre : S3 Unité d'Enseignement : UEF2 Code : ING32

Nombre d'heures d'enseignement (45h sur 15 semaines)

Cours : 1h30/ Semaine TD : 1h30/semaine TP : -

Nombre d'heures de travail personnel pour l'étudiant : 1h30 (22h30 sur 15 semaines)

Nombre de crédits : 5 crédits

Coefficient de la matière : 3

Objectifs de l'enseignement

Ce cours vise la découverte des différentes facettes du processus décisionnel et des modèles et outils associés et la maîtrise de la technologie des Data Warehouses comme application d'un système décisionnel.

Connaissances préalables recommandées

Connaissances générales dans le domaine des Bases de Données et des Systèmes d'Information.

Contenu de la matière :

Chapitre 1 : Introduction aux systèmes décisionnels

Chapitre 2 : Modèles de Système d'Information Décisionnel (SID)

Chapitre 3 : Data Warehouse, Concepts de base et Architecture

Chapitre 4 : Modèles multidimensionnels

Chapitre 5 : Les techniques de Data-Mining (exploration de données)

Mode d'évaluation : Formative et continue

Références (Livres et photocopiés, sites internet, etc).

- W.H. INMON : Building the Data Warehouse

- R.KIMBALL : Concevoir et déployer un Data Warehouse, Guide de conduite de projets. Editions Eyrolles 2000.

- Thomas Connolly, Carolyn Begg: Systèmes de bases de données : approche pratique de la conception, de l'implémentation et de l'administration, Eyrolles 2005
 - J.C. POMEROL, S. BARBA-ROMERO : Multicriterion decision in management : principals and practice. KLUWER ACADEMIC PUBLISHERS 2000.
-

Intitulé du master : Ingénierie du logiciel

Intitulé de la matière : Datamining

Code : DM

Semestre : S3

Unité d'Enseignement : UEF2

Code : ING32

Nombre d'heures d'enseignement (45h sur 15 semaines)

Cours : 1,5 H/ Semaine TD : - TP : 1,5 H/Semaine

Nombre d'heures de travail personnel pour l'étudiant : 1h30 (22h30 sur 15 semaines) Nombre de crédits : 4 crédits

Coefficient de la matière : 2

Objectif

Permettre aux étudiants d'avoir des notions de base sur la statistique décisionnelle et sur les différentes techniques de forage des données.

Connaissances préalables recommandées

Contenu de la matière :

1. Introduction au Data Mining

- Présentation du Data Mining. Différenciation par rapport aux techniques exploratoires des données (Statistique exploratoire, Analyse de Données, etc.).
- Catégorisation des techniques de fouille de données selon les objectifs d'une étude et le type des données.

2. Introduction à l'apprentissage supervisé

- Fondements bayésien de l'apprentissage supervisé. Estimation des probabilités avec le modèle multinomial complet. Élaboration d'une règle d'affectation.
- Évaluation de l'apprentissage, la matrice de confusion et les ratios associés.
- Le principe du partitionnement des données en données « d'apprentissage » et données « test ».

3. Arbres de décision – Introduction

- L'apprentissage par partitionnement. Construction d'un arbre de décision sur un jeu de données synthétique.
- Principaux points à considérer lors de l'induction d'un arbre de décision à partir de données.
- Les trois principales méthodes d'induction d'arbres proposés dans les logiciels. Les différences et les points communs.

4. Arbres de régression

La régression par arbres. Une méthode non linéaire de régression. Rapprochement avec les arbres de décision.

5. Analyse discriminante prédictive

Un modèle paramétrique de discrimination. Analyse discriminante de Fisher. Évaluation globale du modèle, évaluation individuelle des variables.

6. Règles d'association

Construction des règles d'association : items, itemsets fréquents, règles. Les critères supports, confiance, lift.

7. Analyse en Composantes Principales

Construction des axes factoriels. Cercle de corrélations. Projections. Utilisation des variables et individus supplémentaires.

8. Analyse (Factorielle) des Correspondances Multiples

Construction des axes factoriels. Projections et interprétations. Utilisation des variables et individus supplémentaires.

9. Analyse Factorielle des Correspondances

Tableau de contingence. Construction des axes factoriels. Projections et interprétations. Projection des points supplémentaires avec un tableur.

10. Analyse Factorielle Discriminante

Produire des axes factoriels qui permettent de discerner au mieux des groupes d'individus prédéfinis.

11. Arbres de classification

Construire une typologie (des groupes "similaires" d'individus) en utilisant les arbres de classification. Classification automatique. Clustering en anglais. La méthode s'apparente à une généralisation des arbres de décision - arbres de régression.

12. Classification de variables

Classification de variables autour des composantes latentes (VARCLUS). L'idée toujours est de découvrir les principales "dimensions" que recèlent les données. La différence ici est que l'on regroupe les variables (selon leur redondance) plutôt que les individus. Les applications vont au delà de la structuration des données.

12. Régression linéaire

12.1. Pratique de la régression - Diagnostic et sélection de variables

Analyse graphique des résidus. Test de normalité des résidus. Test de Durbin-Watson et des séquences. Détection et traitements des points atypiques et influents. Colinéarité et sélection de variables. Régression sur exogènes qualitative. Détection des ruptures de structure, test de Chow.

12.2. Détection des points atypiques et influents.

Calcul des indicateurs usuels pour la détection des points aberrants et influents dans la régression : Leverage, Résidu Standardisé, Résidu Studentisé, DFFITS, Distance de Cook, COVRATIO, DFBETAS

Mode d'évaluation : Examen final et contrôle *continu*.

Références (*Livres et polycopiés, sites internet, etc*).

Intitulé du master : Ingénierie du logiciel

Intitulé de la matière : Vérification formelle des systèmes **Code : VFS**

Semestre : S3 **Unité d'Enseignement : UEM1** **Code : ING33**

Nombre d'heures d'enseignement (45h sur 15 semaines)

Cours : 1h30/ Semaine TD : - TP : 1h30/ Semaine

Nombre d'heures de travail personnel pour l'étudiant : 1h (15 H sur 15 semaines)

Nombre de crédits : 3 crédits

Coefficient de la matière : 2

Objectifs de l'enseignement

Apprendre à l'étudiant à vérifier et valider un système. La démarche de vérification/validation formelle reste la seule façon de garantir l'absence d'erreur, le problème reste la capacité limitée des outils présents sur le marché qui impliquent l'impossibilité de généraliser cette méthode à 100%.

Connaissances préalables recommandées

Notion de logique et de mathématiques

Contenu de la matière :

- Introduction à la vérification de logiciels
- Classification des techniques de vérification formelles
- Model-checking
 - _ Modélisation (automates, réseaux de Petri, automates temporisés, etc.)
 - _ Spécification de propriétés et logique temporelles
 - _ Algorithmes de vérification
- Model-checking de programmes
- Outils de vérification (Uppaal, Maude & JavaPathfinder)

Evaluation : Examen + Continu

Références

- _ B. Berard et al., Systems and Software Verification: Model-Checking Techniques and Tools, 2001, Springer.
- _ C. Baier and J.P. Katoen, Principles of Model Checking, 2008, The MIT Press.
- _ J.P. Katoen, Concepts, Algorithms, and Tools for Model Checking, 1999, Lecture Notes for Real Time Systems course.
- _ Uppaal : <http://uppaal.org/>
- _ The Maude system : <http://maude.cs.uiuc.edu/>
- _ Java Pathfinder : <http://babelfish.arc.nasa.gov/trac/jpf>

Intitulé du master : Ingénierie du logiciel

Intitulé de la matière : Interface Homme machine

Code : IHM

Semestre : S3

Unité d'Enseignement : UEM1

Code : ING33

Nombre d'heures d'enseignement (45h sur 15 semaines)

Cours : 1h30/ Semaine TD : - TP : 1h30/ Semaine

Nombre d'heures de travail personnel pour l'étudiant : 1h (15 H sur 15 semaines)

Nombre de crédits : 3 crédits

Coefficient de la matière : 2

Objectifs de l'enseignement

Introduire l'étudiant aux différents aspects de l'interface home machine tels les notions élémentaires d'ergonomie, les modèles d'architecture pour les IHM, etc...

Connaissances préalables recommandées

Contenu de la matière :

Introduction

- o Cycle de vie du logiciel interactif
- o Modèles d'architecture pour les IHM (Seeheim, PAC, PAC-Amodeus, un peu de MVC)
- o Catégories d'outils pour la construction des IHM (Logiciels graphiques de base, boîtes à outils / frameworks, générateurs interactifs)
- o Prise en compte des utilisateurs dans le processus de conception des IHM
- o Présentation de l'API Swing de java
- o Méthode d'implémentation efficace du modèle PAC basée sur l'utilisation de design patterns

Mode d'évaluation : Examen + Continu

Evaluation : Examen + Continu

Références

Livres et polycopiés, sites Internet, etc.

Intitulé du master : Ingénierie du logiciel

Intitulé de la matière : Techniques d'expression et communication Code : *TEX*

Semestre : S3 Unité d'Enseignement : **UET1** Code : **ING34**

Nombre d'heures d'enseignement (22h30 sur 15 semaines)

Cours : 1h30/ Semaine TD : - TP : -

Nombre d'heures de travail personnel pour l'étudiant : 1h (15 H sur 15 semaines)

Nombre de crédits : 1 crédits

Coefficient de la matière : 1

Objectifs de l'enseignement

Le but de cet enseignement est d'apprendre aux étudiants la communication orale et écrite, en milieu universitaire scientifique et aussi en milieu social hors de l'université, notamment lors de la recherche d'un premier travail et en entreprise, après le recrutement. Il prépare l'étudiant au monde du travail qu'il rejoindra à la fin de ses études. Si l'étudiant poursuit une carrière universitaire d'enseignant chercheur, il aura acquis dans ce module les connaissances de base de rédaction d'articles de recherches, de réalisation d'une bibliographie, de rapports scientifiques, voire d'ouvrages pédagogiques et de présentation orale de travaux, de communication entre chercheurs etc. Si l'étudiant poursuit une carrière en entreprise, ce module lui aura permis d'apprendre comment se présenter à un concours de recrutement, comment rédiger un curriculum vitae et une lettre de motivation pour le poste recherché, comment réussir un entretien d'embauche etc. et aussi une fois recruté, comment collaborer au travail collectif, comment organiser une équipe de travail, et comment produire les documents internes de l'entreprise (rapports internes, PV de réunion, etc.).

Connaissances préalables recommandées

Cet enseignement est en premier lieu basé sur la connaissance, au moins partielle, de l'anglais. Les étudiants doivent maîtriser certains outils de bureautique pour la rédaction de documents.

Contenu de la matière :

1. Langue : anglais
2. Outils de communication scientifique écrite (Latex, Word, etc.)

3. Techniques de communication écrite

Présentation de méthodes de rédaction de documents différents

- article de recherche
- bibliographie
- ouvrage ou chapitre dans un ouvrage
- rapport interne de recherche
- PV de réunion
- Une demande de recrutement
- Une lettre de motivation
- Curriculum vitæ, etc.

4. Techniques de communication orale

Cette partie devra se faire sous forme d'exercices pratiques où l'étudiant doit communiquer oralement dans les situations (simulées) suivantes:

- Présenter un exposé sur un travail donné,
- se présenter à un groupe de personnes en vue d'un recrutement,
- simuler une réunion de travail, etc.

Intitulé du master : Ingénierie du Logiciel

Intitulé de la matière : éthique et déontologie de travail.

Code : CDT

Semestre : S4

Unité d'Enseignement : UET1

Code : ING 3 4

Nombre d'heures d'enseignement (22h30 sur 15 semaines)

Cours : 1h30/ Semaine TD : - TP : -

Nombre d'heures de travail personnel pour l'étudiant : 1h (15 H sur 15 semaines)

Nombre de crédits : 1 crédits

Coefficient de la matière : 1

Objectifs de l'enseignement : prendre des connaissances aux étudiants sur le rôle de l'université dans le sociaux économique et le rôle de l'enseignant à l'université.

Contenu de la matière

1. Généralités et définition (Morale, Ethique et Déontologie)
2. Les Acteurs de l'Éthique
3. Principes Fondamentaux De L'Éthique
4. Ethique Universitaire
5. L'Université et ses partenaires
6. L'Enseignant à l'Université
7. Les Piliers d'une bonne Université
8. Etablissement de Questionnaire d'évaluation de l'enseignement au département d'informatique

Références bibliographiques

-Charte de l'éthique et de la déontologie universitaires, Alger, mai 2010

Lien (s) :• www.mesrs.dz Source (s) : Ministère de l'Enseignement supérieur/ Etablissement universitaires

- Duquet, Diane, *L'éthique dans la recherche universitaire: une réalité à gérer*, Conseil supérieur de l'Éducation, Québec, 1993

- <http://www.lifl.fr/~duchien/EthiqueRIC2012.pdf>

