

Semestre 3

| Unité d'Enseignement | VHS | V.H hebdomadaire | | | | Coeff | Crédits | Mode d'évaluation | |
|---|--------|------------------|-------------|-------------|--------------|-----------|-----------|-------------------|--------|
| | 15 sem | C | TD | TP | Autres | | | Continu | Examen |
| UE fondamentales | | | | | | | | | |
| UEF1(O/P) | | 4h30 | 3h00 | 3h00 | 6h00 | 6 | 11 | | |
| Architecture des ordinateurs | | 1h30 | 1h30 | 1h30 | 3h00 | 3 | 5 | 40% | 60% |
| Algorithmique et structure de données 3 | | 3h00 | 1h30 | 1h30 | 3h00 | 3 | 6 | 40% | 60% |
| UEF2(O/P) | | 3h00 | 3h00 | 1h30 | 6h00 | 5 | 9 | | |
| Systèmes d'information | | 1h30 | 1h30 | 1h30 | 3h00 | 3 | 5 | 40% | 60% |
| Théorie des graphes | | 1h30 | 1h30 | | 3h00 | 2 | 4 | 40% | 60% |
| UE méthodologie | | | | | | | | | |
| UEM (O/P) | | 3h00 | 1h30 | 1h30 | 06h00 | 4 | 8 | | |
| Méthodes Numériques | | 1h30 | | 1h30 | 3h00 | 2 | 4 | 40% | 60% |
| Logique Mathématique | | 1h30 | 1h30 | | 3h00 | 2 | 4 | 40% | 60% |
| Unité Transversale | | | 1h30 | | 2h00 | 1 | 2 | | |
| Langue Etrangère | | | 1h30 | | 2h00 | 1 | 2 | | 100% |
| Total Semestre 3 (25h30) | | 10h30 | 9h00 | 6h00 | 20h00 | 16 | 30 | | |

Semestre 4

| Unité d'Enseignement | VHS | V.H hebdomadaire | | | | Coeff | Crédits | Mode d'évaluation | |
|---------------------------------|--------|------------------|-------------|-------------|--------------|-----------|-----------|-------------------|--------|
| | 15 sem | C | TD | TP | Autres | | | Continu | Examen |
| UE fondamentales | | | | | | | | | |
| UEF1(O/P) | | 3h00 | 3h00 | 1h30 | 6h00 | 5 | 10 | | |
| Théorie des langages | | 1h30 | 1h30 | | 3h00 | 2 | 5 | 40% | 60% |
| Système d'exploitation 1 | | 1h30 | 1h30 | 1h30 | 3h00 | 3 | 5 | 40% | 60% |
| UEF2(O/P) | | 4h30 | 1h30 | 3h00 | 6h00 | 6 | 10 | | |
| Bases de données | | 1h30 | 1h30 | 1h30 | 3h00 | 3 | 5 | 40% | 60% |
| Réseaux | | 3h00 | | 1h30 | 3h00 | 3 | 5 | 40% | 60% |
| UE méthodologie | | | | | | | | | |
| UEM (O/P) | | 3h00 | | 3h00 | 6h00 | 4 | 8 | | |
| Programmation orientée objet | | 1h30 | | 1h30 | 3h00 | 2 | 4 | 40% | 60% |
| Développement Applications Web | | 1h30 | | 1h30 | 3h00 | 2 | 4 | 40% | 60% |
| Unité Transversale | | | 1h30 | | 2h00 | 1 | 2 | | |
| Langue Etrangère | | | 1h30 | | 2h00 | 1 | 2 | | 100% |
| Total Semestre 4 (24h00) | | 10h30 | 6h00 | 7h30 | 20h00 | 16 | 30 | | |

Semestre : 03
Unité d'enseignement fondamentale : UEF1
Matière : Architecture des ordinateurs
Crédits : 5
Coefficient : 3

Objectifs de l'enseignement : la matière a pour objectif de mettre en clair le principe de fonctionnement de l'ordinateur avec une présentation détaillée de l'architecture de l'ordinateur.

Connaissances préalables recommandées :

Contenu de la matière :

Chapitre 1 :

- Introduction à la notion d'architecture des ordinateurs
- La machine de Von Neumann et la machine Harvard.

Chapitre 2 : Principaux composants d'un ordinateur

- Schéma global d'une architecture
- L'UAL
- Les bus
- Les registres
- La mémoire interne : mémoire RAM (SRAM et DRAM), ROM, temps d'accès, latence,...
- La mémoire cache : utilité et principe, algorithmes de gestion du cache (notions de base)
- Hiérarchie de mémoires

Chapitre 3 : Notions sur les instructions d'un ordinateur :

- Langage de haut niveau, assembleur, langage machine
- Les instructions machines usuelles (arithmétiques, logiques, de comparaison, chargement, rangement, transfert, sauts,...)
- Principe de compilation et d'assemblage (notions de base)
- L'unité de contrôle et de commande
- Phases d'exécution d'une instruction (Recherche, décodage, exécution, rangement des résultats)
- UCC pipeline
- L'horloge et le séquenceur

Chapitre 4 : Le processeur

- Rôle du processeur, calcul de CPI (Cycle per Instruction), les processeurs CISC et RISC.

- Le microprocesseur MIPS R3000
- Structure externe du processeur MIPS R3000
- Structure interne du processeur MIPS R3000
- Jeu d'instructions, Formats et programmation du MIPS R3000.
- Programmation du MIPS R3000

Chapitre 5 : instructions spéciales

Notions sur les interruptions, les entrées-sorties et les instructions systèmes (cas du MIPS R3000)

Travaux dirigés

TD concernant le chapitre 1 : (Exercices de rappels sur la représentation de l'information et l'algèbre de Boole)

TDs concernant le chapitre 2 :

- Etude et utilisation d'une UAL (exemple l'UAL 74181).
- Exercices sur les registres (entrées-sorties d'un registre, signaux intervenant sur un registre : chargement, sortie (sorties en 3-états), registres à chargement parallèle, registres à décalage.
- Exercices sur les bus : Mécanismes de gestion de conflits d'accès simultanés.
- Les mémoires : exercices sur les mémoires RAM : formats des adresses et des données, signaux de lecture et d'écriture (chronogrammes), calcul de vitesses d'accès en mémoire.
Les mémoires mortes.
Organisation des mémoires : plan de décodage pour l'augmentation des mots et de la capacité mémoire.
- La mémoire cache : Exercices portant sur la mémoire cache (organisation, défauts de cache, remplacements de lignes)

TD portant sur les phases d'exécution d'une instruction :

(Pour une meilleure compréhension de ce Td, il serait intéressant de présenter une architecture très simple avec un compteur ordinal, un registre d'adresse mémoire, une mémoire RAM, un registre de données mémoire, un registre instruction, une batterie de registres à usage général, une UAL, deux registres en entrée de l'UAL, un accumulateur, un registre code conditions, un ensemble de bus internes un séquenceur et une horloge de base)

Prendre deux instructions simples (exemples : Addition et branchement pour en voir le déroulement)

Phases :

- Recherche de l'instruction.
- Décodage de l'instruction.
- Recherche des opérandes éventuels.
- Exécution de l'instruction.
- Rangement des résultats éventuels.

TD portant sur l'exécution sur une unité pipeline

- Exécution pipeline
- Gain

TD portant sur le microprocesseur MIPS R3000

- Détermination des 3 formats (Registre, immédiat, Jump)
- Codage d'instructions.
- Décodage d'instructions
- Etude de quelques instructions

- Réalisation de petits programmes assembleur MIPS

TD portant sur des instructions spéciales telles qu'appel de superviseur et interruptions

Mode d'évaluation : Examen (60%) , contrôle continu (40%)

Références

1. Alain Cazes , Joëlle Delacroix, Architecture des machines et des systèmes informatiques 4 ème édition, *Collection : Informatique, Dunod, 2011.*
2. [Andrew S. Tanenbaum](#), [Todd Austin](#) Structured Computer Organization, Pearson, 2012.
3. [Paolo Zanella](#), [Yves Ligier](#), [Emmanuel Lazard](#), Architecture et technologie des ordinateurs : Cours et exercices - *Collection : Sciences Sup, Dunod*, 5ème édition, 2013.
Liens vers le microprocesseur MIPS R3000
4. <ftp://132.227.86.9/pub/mips/mips.asm.pdf>
5. <ftp://asim.lip6.fr/pub/mips/mips.externe.pdf>
6. <ftp://asim.lip6.fr/pub/mips/mips.interne.pdf>

Semestre : 03

Unité d'enseignement fondamentale : UEF1

Matière Algorithmique et structure de données 3

Crédits : 6

Coefficient : 3

Objectifs de l'enseignement : ce module permettra aux étudiants d'apprendre d'une part la mise au point de certain algorithme de base en informatique, d'autre part, ils apprendront à manipuler des structure de données plus développées.

Connaissances préalables recommandées : algorithmique de base

Contenu de la matière :

Rappel

Chapitre 1 : Complexité algorithmique

1. Introduction à la complexité
2. Calcul de complexité

Chapitre 2 : Algorithmes de tri

1. Présentation
2. Tri à bulles
3. Tri par sélection
4. Tri par insertion
5. Tri fusion
6. Tri rapide

Chapitre 3 : Les arbres

1. Introduction
2. Définitions
3. Arbre binaire
 - 3.1. Définition
 - 3.2. Passage d'un arbre n-aire à arbre binaire
 - 3.3. Représentation chaînée d'un arbre binaire
 - 3.4. Parcours d'un arbre binaire
 - 3.4.1. *Parcours préfixé (préordre ou RGD)*
 - 3.4.2. *Parcours infixé (projectif, symétrique ou encore GRD)*
 - 3.4.3. *Parcours postfixé (ordre terminal ou GDR)*
 - 3.5. Arbres binaires particuliers
 - 3.5.1. *Arbre binaire complet*
 - 3.5.3. *Arbre binaire de recherche*

Chapitre 4 : Les graphes

1. Définition
2. Représentation des graphes
3. Parcours des graphes

NB : TP en C.

Mode d'évaluation : Examen (60%) , contrôle continu (40%)

Références

Semestre : 03

Unité d'enseignement Fondamentale : UEF2

Matière : Systèmes d'information

Crédits : 5

Coefficient : 3

Objectifs de l'enseignement : Comprendre ce qu'est un système d'information d'entreprise ; (2) Comprendre les différentes dimensions constitutives d'un SI : a. Dimension technique b. Dimension organisationnelle c. Dimension managériale (3) Comprendre les différents éléments d'un SI : a. système pilotage b. système décisionnel c. système opérationnel (4) Comprendre l'articulation du SI avec la stratégie d'entreprise (gouvernance des SI – gestion des projets SI)

Connaissances préalables recommandées : algorithmique,

Contenu de la matière :

Chapitre 1 : Généralité

- Définitions et caractérisations de l'entreprise (les aspects fonctionnels et structurels),
- Approche systémique des organisations : Présentation globale des trois systèmes (le système de décision, classification des décisions: par niveau et par méthode, une technique de décision programmable, les tables de décision), Le système d'information (Aspects fonctionnels et Aspects structurels : notion de station, poste de travail, de flux, documents), Le diagramme de flux.

Chapitre 2 : Les Techniques de représentation de l'information

Notion d'information, Formes et manipulation de l'information, Etude de l'information : Classe et réalisation de classe, description de classe, ... Schéma et codification de l'information

Chapitre 3 : Saisie et contrôle de l'information

Différents types de contrôle de l'information

Chapitre 4 : Méthodologie de développement d'un SI: MERISE

- Processus de développement d'un SI
- Niveau d'abstraction de modèle de données et des traitements
- Méthodologie MERISE
- Concepts pour la modélisation statique (Notion d'entité et d'association, un modèle conceptuel de données : le MCD de MERISE (à comparer avec le diagramme de classes UML),
- Concepts pour la modélisation dynamique : MCT de MERISE (à comparer avec les diagrammes dynamiques d'UML)

Mode d'évaluation : Examen (60%) , contrôle continu (40%)

Références

1. « Systèmes d'information organisationnels », Coord. P. Vidal, P. Planeix, 2005.
2. « Présents et futurs des systèmes d'information », Coord. M-L. Caron-Fasan & N. Lesca, 2003, PUG, 323 p.
3. « Le e-management. Quelles transformations pour l'entreprise ? », Kalika M. & alii, 2003, Editions Liasons, 192 p.
4. J.L.Lemoigne, La théorie du système général. PUF- V. Bertalanfy, Théorie générale des systèmes. Dunod. X. Castellani, Méthode générale d'analyse d'une application informatique.- Tardieu et al. , « la méthode merise : principes et outils », éd. d'organisation, 1983.- Tardieu et al. , « la méthode merise : démarche et pratique » éd. d'organisation, 1985.- Tabourier, « de l'autre côté de Merise », éd. d'organisation, 1986.- J. P. Mathéron, « Comprendre Merise », 1990

Semestre : 03

Unité d'enseignement Fondamentale : UEF1

Matière : Théories des graphes

Crédits : 4

Coefficient : 2

Objectifs de l'enseignement : Les théories des graphes sont devenues un fondement théorique et pratique incontournable dans le processus de modélisation de certains problèmes dans plusieurs domaines. L'apport des graphes dans la résolution des problèmes réside dans la simplicité graphique, la similitude avec des aspects distribués et les notions de parcours et de recherches de chemins. L'objectif de ce cours est de présenter à l'étudiant d'une part un de modélisation de solution sous forme de graphe, d'autre part ce cours contiendra un ensemble de techniques permettant à l'étudiant de résoudre ses problèmes à travers des algorithmes comme la recherche de chemin minimal, le flot maximal etc.

Connaissances préalables recommandées

Contenu de la matière :

Chapitre I. Définitions de base

- I.1. Définition "intuitive" d'un graphe
2. Définition mathématique d'un graphe
3. Ordre, orientation et multiplicité
 - 3.1. Ordre
 - 3.2. Orientation
 - 3.3. Multiplicité
4. Relations entre les éléments d'un graphe
 - 4.1 Relations entre sommets
 - 4.2 Relations entre arcs et sommets
 - 4.3 Qualificatifs des graphes
5. Matrices associées à un graphe
 - 5.1 Matrice d'incidence sommet-arc
 - 5.2 Matrice d'adjacence ou d'incidence sommets-sommets
 - 5.3 Forme condensée des matrices creuses
6. Vocabulaire lié à la connexité
 - 6.1 Chaîne, chemin, longueur
 - 6.2 Connexité
 - 6.3 Cycle et circuit
 - 6.4 Cocycle et cocircuit.

Chapitre II. Cycles

1. Nombres cyclomatique et cocyclomatique
 1. Décomposition des cycles et des cocycles en sommes élémentaires
 2. Lemme des arcs colorés (Minty 1960)
 3. Base de cycles et base de cocycles
2. Planarité

1. Graphe Planaire
2. Formule d'Euler
3. Théorème de Kuratowski (1930)
4. Graphe Dual

3. Arbre, forêt et arborescence

1. Définitions
2. Propriétés
3. Arbre maximal (ou couvrant).

Chapitre III. Flots

1. Définitions
2. Recherche d'un flot maximum dans un réseau de transport
 4. Définition
 5. Théorème de Ford-Fulkerson
 6. Algorithme de Ford-Fulkerson
3. Recherche d'un flot compatible

Chapitre IV. Problèmes de cheminement

1. Recherche des composantes connexes
 1. Présentation des objectifs
 2. Algorithme de Trémaux-Tarjan
2. Recherche du plus court chemin
 1. Présentation des conditions
 2. Algorithme de Moore-Dijkstra
3. Recherche d'un arbre de poids extrémum
 1. Présentation des objectifs
 2. Algorithme de Kruskal 1956

Chapitre V. Problèmes Hamiltonien et Eulérien

1. Problème Hamiltonien
 1. Définitions
 2. Condition nécessaire d'existence d'un cycle hamiltonien
 3. Condition suffisante d'existence d'un circuit hamiltonien
 4. Condition suffisante d'existence d'un cycle hamiltonien
2. Problème Eulérien
 1. Définitions
 2. Condition nécessaire et suffisante d'existence d'une chaîne eulérienne
 3. Algorithme local pour tracer un cycle eulérien
 4. Lien entre problème eulérien et hamiltonien

Chapitre VI. Coloration

1. Définitions
2. Coloration des sommets
3. Coloration des arêtes

4. Propositions
5. Le théorème des "4 couleurs"
6. Graphe parfait

Mode d'évaluation : Examen (60%) , contrôle continu (40%)

Références

1. Claude Berge, Graphes et hypergraphes, Bordas 1973, (300 pages).
2. Nguyen Huy Xuong, Mathématiques discrètes et informatique, Masson, 1997
3. Aimé Sache, La théorie des graphes, Que-Sais-Je ?, 1974 ; réédition prévue en 2004 chez Cassini.
4. M. Kaufmann, Des points des flèches, la théorie des graphes, Dunod, Sciencespoche, épuisé.
5. Alan Gibbons, Algorithmic graph theory, Cambridge University Press, 1985
6. Reinhard Diestel, Graph Theory, Second Edition, Springer-Verlag, 2000.
7. Bojan Mohar, Carsten Thomassen, Graphs on surfaces, John Hopkins University Press, Baltimore, 2001.

Semestre : 03
Unité d'enseignement Méthodologique : UEM
Matière : Méthodes numériques
Crédits : 4
Coefficient : 2

Objectifs de l'enseignement : cette matière permettra aux étudiants d'investir le domaine des méthodes numériques nécessaires à la résolution des problèmes

Connaissances préalables recommandées : mathématiques de base

Contenu de la matière :

Chapitre 1 : Généralités sur l'analyse numérique et le calcul scientifique

- 1.1 Motivations.
- 1.2 Arithmétique en virgule flottante et erreurs d'arrondis
 - 1.2.1 Représentation des nombres en machine
 - 1.2.2 Erreurs d'arrondis
- 1.3 Stabilité et analyse d'erreur des méthodes numériques et conditionnement d'un problème

Chapitre 2 : Méthodes directes de résolution des systèmes linéaires

- 2.1 Remarques sur la résolution des systèmes triangulaires
- 2.2 Méthode d'élimination de Gauss
- 2.3 Interprétation matricielle de l'élimination de Gauss : la factorisation LU

Chapitre 3 : Méthodes itératives de résolution des systèmes linéaires

- 3.1 Généralités
- 3.2 Méthodes de Jacobi et de sur-relaxation
- 3.3 Méthodes de Gauss-Seidel et de sur-relaxation successive
- 3.4 Remarques sur l'implémentation des méthodes itératives
- 3.5 Convergence des méthodes de Jacobi et Gauss-Seidel

Chapitre 4 : Calcul de valeurs et de vecteurs propres

- 4.1 Localisation des valeurs propres
- 4.2 Méthode de la puissance

Chapitre 5 : Analyse matricielle

- 5.1 Espaces vectoriels
- 5.2 Matrices
 - 5.2.1 Opérations sur les matrices
 - 5.2.2 Liens entre applications linéaires et matrices
 - 5.2.3 Inverse d'une matrice
 - 5.2.4 Trace et déterminant d'une matrice
 - 5.2.5 Valeurs et vecteurs propres
 - 5.2.6 Matrices semblables

- 5.2.7 Quelques matrices particulières
- 5.3 Normes et produits scalaires
 - 5.3.1 Définitions
 - 5.3.2 Produits scalaires et normes vectoriels
 - 5.3.3 Normes de matrices

Mode d'évaluation : Examen (100%)

Références :

1. M. Schatzman Analyse numérique : une approche mathématique, Dunod 2004.
2. P.G. Ciarlet, Introduction à l'analyse matricielle et à l'optimisation, Masson 1990.
3. J. Demmel, Applied Numerical Linear Analysis, SIAM 1997 ;
4. C. D. Meyer, Matrix Analysis and Applied Linear Algebra, SIAM 2000 ;
5. P. Lascaux et J. Théodor, Analyse numérique matricielle appliquée à l'art de l'ingénieur, 2 tomes, Masson 1988.
6. G. H. Golub, C. F. van Loan, Matrix Computations, The Johns Hopkins University Press, 1989.
7. W.H. Press, B.P. Flannery, S.A. Teukolsky et W.T. Ve

Semestre : 03
Unité d'enseignement méthodologique : UM
Matière : Logique mathématique
Crédits : 4
Coefficient : 2

Objectifs de l'enseignement : Formalisation du raisonnement humain

Connaissances préalables recommandées : connaissances de base en mathématiques et du bon sens

Contenu de la matière :

Chapitre 1 : Introduction

- a. Objets de la logique
- b. Syntaxe et sémantique

Chapitre 2 : Logique des propositions

- i. Syntaxe
 - 1. Les propositions
 - 2. Les connecteurs logiques
 - 3. Variables et formules propositionnelles
 - 4. Substitution dans une formule
 - 5. Formules logiques et arbres
- ii. Sémantique
 - 1. Interprétation
 - 2. Tables de vérité
 - 3. Tautologies et antilogies
 - 4. Equivalence sémantique
 - 5. Formes normales conjonctives et disjonctives
 - 6. Satisfaisabilité et validité
- iii. Résolution
 - 1. Réfutation
 - 2. Mise en forme clausale
 - 3. Règle de résolution propositionnelle
 - 4. La méthode de résolution propositionnelle

Chapitre 3: Logique des prédicats

- c. Syntaxe
 - i. Termes
 - ii. Prédicats
 - iii. Quantificateurs

- iv. Formules
 - 1. Portée d'un identificateur
 - 2. Variables libres, variables liées
- d. Sémantique
 - i. Structure
 - ii. Satisfaction d'une formule

Mode d'évaluation : Examen (60%) , contrôle continu (40%)

Références

1. S.C. Kleene. Logique mathématique. Collection U, Armand Colin, Paris 1971.
2. J.L. Krivine. Elements of Mathematical Logic. North-Holland Publishing Company Amsterdam.
3. R. Cori. Logique mathématique. Tome 1 : Calcul propositionnel, Algèbre de Boole, calcul des prédicats. Dunod.

Semestre : 03

Unité d'enseignement Transversale : UT

Matière : Langue étrangère

Crédits : 2

Coefficient : 1

Objectifs de l'enseignement : L'objectif de ce module est d'apprendre à l'étudiant les approches de synthèse de document et d'extraire les connaissances importantes. Il permettra à l'étudiant de développer ses capacités pour la rédaction de texte etc.

Connaissances préalables recommandées : maitrise de la syntaxe de langue anglaise

Contenu de la matière :

Semestre : 04
Unité d'enseignement fondamentale : UEF1
Matière : Théorie des langages
Crédits : 5
Coefficient : 2

Objectifs de l'enseignement : comprendre la théorie et les outils de la théorie des langages

Connaissances préalables recommandées : connaissances de base en mathématiques et en informatique
/

Contenu de la matière :

Chapitre 1 : Introduction (objectifs ...)

Chapitre 2 : Alphabets, Mots, Langages

Chapitre 3 : Grammaires

1. Définitions
2. Dérivation et langage engendré
3. Arbre de dérivation
4. Hiérarchie de Chomsky

Chapitre 4: Automates d'états finis (AEF)

1. AEF déterministes
2. Représentations d'un automate
3. Automates équivalents et complets
4. AEF non déterministes (déterminisation)
5. Automates et langages réguliers (transformations et propriétés))

Chapitre 5: Expressions Régulières

1. Définitions
2. Théorème de Kleene
3. Lemme de l'étoile

Chapitre 6: Minimisation d'un AEF

Chapitre 7: Langages Algébriques

1. Propriétés d'une grammaire régulière
2. Transformations d'une grammaire
3. Grammaire réduite
4. Grammaire propre
5. Elimination des récursivités à gauche
6. Formes normales

Chapitre 8: Automates à Piles

1. Définition
2. Configuration, transition et calcul
3. Critères d'acceptation

4. Automates à piles déterministes

Chapitre 9: Machine de Turing

1. Définition
2. Configuration, transition et calcul
3. Acceptation

Mode d'évaluation : Examen (60%) , contrôle continu (40%)

Références

1. P. Wolper. Introduction à la calculabilité. 2006, Dunod.
2. P. Séébold. Théorie des automates. 2009, Vuibert.
3. J.M. Autebert Théorie des langages et des automates. 1994, Masson.
4. J. Hopcroft, J. Ullman. Introduction to Automata Theory, Languages and Compilation 1979, Addison-Wesley

Semestre : 04
Unité d'enseignement fondamentale : UEF1
Matière : Système d'exploitation 1
Crédits : 5
Coefficient : 3

Objectifs de l'enseignement :

Connaissances préalables recommandées :
/

Contenu de la matière :

Chapitre 1 : Introduction

- Notion de système d'exploitation.
- Fonctions et rôles.
- Exemples de systèmes d'exploitation (Windows, Unix, Android,...)

Chapitre 2 : La gestion du processeur

- Définitions
 - Notion de Programme.
 - Notion de Processus.
 - Notion de Thread.
 - Notion de ressource
 - Notion de travail (Job)
- Différents états d'un processus.
- Hiérarchies de processus.
- Relations entre processus (compétition, coopération et synchronisation).
- Techniques d'ordonnancement de processus :
 - Critères (Equité, efficacité, temps de réponse, temps d'exécution, rendement)
- Algorithmes d'ordonnancement (parmi les plus utilisés) :
 - Tourniquet (Round Robin RR).
 - Algorithme du premier entré, premier servi ou FCFS (First Come First-Served).
 - Algorithme du travail le plus court d'abord ou SJF (Shortest Job First).
 - Algorithme du temps restant le plus court ou SRT (Shortest Remaining Time).
 - Algorithme avec priorité.

Chapitre 3 : Gestion de la mémoire

- Objectifs.
- Monoprogrammation.
- Multiprogrammation :
 - a) Partitions multiples contiguës.
 - Partitions contiguës fixes.
 - Partitions contiguës dynamiques:

- 1- Stratégie du premier qui convient (First Fit)
- 2- Stratégie du meilleur qui convient (Best Fit)
- 3- Stratégie du pire qui convient (Worst Fit).
 - a. Partitions contiguës Siamois (Buddy system)
 - b. Ré-allocation et protection
 - c. Va et vient (Swap)
 - d. Fragmentation et Compactage
- Multiprogrammation et partitions multiples non contiguës
 - 1. Pagination
 - 2. Segmentation
 - 3. Segmentation paginée.
- La mémoire virtuelle
 - Concept de mémoire virtuelle.
 - Overlays (segments de recouvrement)
 - Pagination à la demande
 - Quelques algorithmes de remplacement des pages :
 - Algorithme optimal
 - Remplacement Aléatoire
 - Ordre Chronologique de Chargement (FIFO) (avec remarque sur l'anomalie de Belady).
 - Ordre Chronologique d'utilisation (LRU: Least Recently Used).
 - Fréquence d'utilisation (LFU: Least Frequently Used).
 - Algorithme de la seconde chance.

Chapitre 4 : Le système Unix.

Travaux dirigés et pratiques

Les TDs porteront sur propositions d'algorithmes autour des différents chapitres. Ces algorithmes seront développés en TP en utilisant le langage au langage C sous Unix. Le système Unix fera l'objet des premières séances de TPs.

Mode d'évaluation : Examen (60%) , contrôle continu (40%)

Références

1. Tanenbaum, Modern operating systems, thirteenth edition, Pearson, 2014
2. A. Tanenbaum, Systèmes d'exploitation, Dunod, 1994.
3. [Michel Divay, Unix, Linux et les systèmes d'exploitation : cours et exercices corrigés](#) -, 2004.
4. Crocus, Systèmes d'exploitation des ordinateurs, 1993.
5. [Sacha Krakowiak](#), Principes des systèmes d'exploitation des ordinateurs, Dunod, 1993

Semestre : 4
Unité d'enseignement fondamentale: UEF2
Matière : Bases de Données
Crédits : 5
Coefficient : 3

Objectifs de l'enseignement : Ce cours devrait permettre à l'étudiant d'identifier l'intérêt de structurer et manipuler les données sous forme tabulaire. A travers le modèle relationnel et l'algèbre relationnelle sous-jacente orientés plus vers l'aspect pratique, l'étudiant devrait comprendre l'importance de structurer les données, le concept d'indépendance des données et des traitements, ainsi que l'intégrité et la cohérence des données.

Connaissances préalables recommandées : L'étudiant est sensé comprendre ce que c'est des fichiers (textes, binaires ou typés) et les avoir créés avec les langages préalablement étudiés.

Contenu de la matière :

Chapitre 1 : Présentation des bases de données

1. Notions de fichiers (intérêts et limites)
2. Définition de base de données
3. Définition de système de gestion de base de données
4. Types de modèles de données (sémantique, entité-association, hiérarchique, réseau, relationnel)

Chapitre 2 : Modèle relationnel

1. Définition du modèle relationnel
2. Concepts de base (Attribut, Tuple, Domaine, Relation)
3. Schéma de relation
4. Normalisation
 - a. Clé de relation et dépendance fonctionnelle (Clé primaire et clé étrangère)
 - b. Contraintes d'intégrité
 - c. Formes normales (1FN, 2FN, 3FN, FN de Boyce-Codd)
5. Schéma de base de données
- Modèle relationnel logique (SQL)
 - a. Table, colonne, et ligne
 - b. Description de SQL (Structured Query Language)
 - c. Définitions de données
 - i. Création de table (CREATE)
 - ii. Modification de schéma (ALTER, DROP)
 - d. Manipulation des données (INSERT, UPDATE, DELETE)

Chapitre 3 : Algèbre relationnelle

1. Définition
2. Opérations et opérateurs unaires
 - a. Sélection
 - b. Projection
 - c. Traduction en SQL
 - i. Requêtes simples (SELECT-FROM)
 - ii. Sélection de colonne (clause WHERE)

- iii. Tri de résultats (ORDER BY)
- 3. Opérations et opérateurs ensemblistes
 - a. Union
 - b. Intersection
 - c. Différence
 - d. Produit cartésien
 - e. Jointure (Thêta, naturelle, équijointure, externe)
 - f. Division
 - g. Traduction en SQL
 - i. Opérateurs d'union, d'intersection, et de différence
 - ii. Produit cartésien (sans jointure)
 - iii. Jointure de tables (condition de jointure)
 - iv. Fonctions d'agrégat
 - v. Clause GROUP BY ... HAVING

Mode d'évaluation : Examen (60%) , contrôle continu (40%)

Références:

- 1. Bases de données. Georges Gardarin. 5^{ème} édition 2003
- 2. SQL Les fondamentaux du langage. Eric Godoc et Anne-Christine Bisson. Edition Eni. 2017
- 3. Bases de données : concepts, utilisation et développement. Jean-Luc Hainaut. Édition DUNOD. 2015

Semestre : 04
Unité d'enseignement fondamentale : UEF2
Matière : Réseaux
Crédits : 5
Coefficient : 3

Objectifs de l'enseignement : cette matière a pour objectif de donner aux étudiants les notions indispensables pour une bonne compréhension des réseaux. Ils doivent être capable d'expliquer ce qu'est un réseau, de quoi il se compose, comment des ordinateurs peuvent communiquer entre eux, décrire les différents types de médias, les différents types de topologies ainsi qu'une étude détaillée sur les cinq couches du modèle Internet.

- Rendre l'étudiant apte à comprendre le fonctionnement, à planifier l'installation et à utiliser un réseau d'ordinateurs.
- Familiariser l'étudiant aux diverses couches d'implantation d'un réseau d'ordinateurs.
- Initier l'étudiant aux principaux protocoles de communication et de routage des messages.
- Familiariser l'étudiant avec les principales composantes d'un réseau d'ordinateurs.
- Rendre l'étudiant apte à utiliser les services de base d'un réseau à l'intérieur d'un programme.

Connaissances préalables recommandées :

/

Contenu de la matière :

Chapitre I : Introduction au Réseaux

- Usage des réseaux
- Caractéristiques Physiques
- Topologies des réseaux
- Modèles de références (OSI, TCP/IP)
- Types de passerelles

Chapitre II: Couche Physique

- Terminologie de Réseaux
- Signaux, décomposition, bruit
- Supports de transmission guidés et non-guidés
- Transmission digitale : Conversion de l'analogue au digital
- Transmission digitale : Conversion du digital au digital
- Echantillonnage
- Transmission analogue : Conversion du digital à l'analogue
- Transmission analogue : Conversion de l'analogue à l'analogue
- Multiplexeur et Concentrateur

Chapitre III: Couche Liaison de Données

- Adressage
- Control de flux
- Norme 802.3 et format Ethernet
- Contrôle des erreurs
- Contrôle d'Accès multiple
- Commutation de circuit

Chapitre IV: Couche Réseaux

- Adressage IP, classes, notion des sous-réseaux
- Protocole IP : IPV4, IPV6
- Fragmentation des paquets
- Commutation de paquets
- Routage : techniques centralisées, techniques distribuées
- Routage statique et routage dynamique
- Routage hiérarchique et externe

Chapitre IV: Couche Transport

- Notion d'adresse transport
- Protocoles UDP et TCP
- Qualité de service
- Control de congestion

Chapitre IV: Couche Application

- Protocole SMTP
- Protocole HTTP
- Protocole FTP
- Protocole DHCP
- Protocole DNS

Travaux pratiques

TP 1 : Configuration de base d'un réseau

TP 2 : Programmation réseau (Socket)

TP 3 : Routage

TP 4 : Analyseur de protocoles

Mode d'évaluation : Examen (60%) , contrôle continu (40%)

Références

1. Forouzan, Behrouz A., and S. C. Fegan. "Data communication and computer networks." (2007).
2. Tanenbaum, Andrew S. "Computer networks, 4-th edition." ed: Prentice Hall (2003).

Semestre : 4
Unité d'enseignement méthodologique: UEM
Matière : Programmation Orientée Objet
Crédits : 4
Coefficient : 2

Objectifs de l'enseignement : L'objectif de ce cours est l'introduction des concepts de base de la programmation orientée objet (POO) par la pratique en utilisant le langage Java. Chaque chapitre comporte certaines notions qui sont traduites à sa fin en Java pour que l'étudiant puisse traduire les concepts théoriques acquis en pratique. A la fin du semestre, l'étudiant est censé avoir acquis les compétences suivantes :

- 1- L'essence de la programmation objet et sa transformation en langage Java
- 2- Acquérir un raisonnement intuitif pour donner une solution à un problème simple selon l'approche orientée objet
- 3- Ecrire un programme en langage Java qui soit fonctionnel
- 4- L'essence et l'importance du raisonnement OO et de la POO

Connaissances préalables : Langage C

Contenu de la matière :

Chapitre 1. Bases de la POO

1. Introduction
2. Concepts fondamentaux de la POO
 - a. Petit historique de la POO
 - b. Programmation procédurale vs programmation par objets
 - c. Réutilisation de code
 - d. Introduction à la modularité
3. Les d'objets et les classes
 - a. Notions d'objet
 - b. Notions de classe
 - c. Attributs
 - d. Notion de message
 - e. Résolution de problèmes par échange de messages
4. Introduction à Java
 - a. Types et structures de contrôle en Java
 - b. Classes et instanciation
 - c. Méthodes
 - d. Les références et passage de paramètres
 - e. Entrées/Sorties
 - f. Constructeur par défaut et autres constructeurs
 - g. Destructeurs

Chapitre 2. Encapsulation

1. Niveaux de visibilité
2. Encapsulation
 - a. Encapsulation de données (attributs)

- b. Encapsulation de code (Messages)
- 3. Encapsulation en Java
 - a. Contrôle d'accès (public, private)
 - b. Accesseurs (get et set)
 - c. Accès à l'instance (this)
 - d. Variables et méthodes de classe (static)

Chapitre 3. Héritage

- 1. Sous-classes et héritage
- 2. Héritage simple, héritage multiple
- 3. Hiérarchie de classes
- 4. Polymorphisme
- 5. Héritage et polymorphisme en Java
 - a. Héritage simple (extends)
 - b. Encapsulation dans l'héritage
 - i. Protection des membres (protected)
 - ii. Constructeurs des classes (this(), super())
 - iii. Classe 'Object'
 - iv. Transtypage implicite et explicite
 - v. Limitation de l'héritage (final)
 - c. Polymorphisme
 - i. Surcharge de méthodes
 - ii. Redéfinition de méthodes
 - d. Classes abstraites (utilisation et importance)
 - e. Interfaces (utilisation et importance)

Mode d'évaluation : Examen (60%) , contrôle continu (40%)

Références

- 1. Apprendre la Programmation Orientée Objet avec le langage Java. Luc Gervais. Eni. 2^{ème} édition.
- 2. <https://openclassrooms.com/courses/apprenez-a-programmer-en-java>
- 3. Java 8 - Apprendre la Programmation Orientée Objet et maîtrisez le langage. Thierry GROUSSARD Luc GERVAIS. Edition ENI. 2015.
- 4. La programmation objet en Java. Michel Divay. Edition DUNOD. 2006.

Semestre : 04
Unité d'enseignement méthodologique : UEM
Matière : Développement Applications Web
Crédits : 4
Coefficient : 2

Objectifs de l'enseignement : L'ultime objectif est d'apprendre à mettre en œuvre une application Web.

Connaissances préalables recommandées : Notions fondamentales sur l'algorithme et la programmation. Notions de base sur Internet et Réseaux.

Contenu de la matière :

Chapitre 1: Introduction au World Wide Web

1. Définition et historique
2. Architecture Client/Serveur
3. Protocole HTTP.

Chapitre 2 : Langages de programmation pour le Web

1. Généralités : page statique, page dynamique et applications Web
2. Langages de balise : définition et historique
3. HTML
 - 3.1. Qu'est ce que le HTML ?
 - 3.2. Contexte d'exécution HTML
 - 3.3. HTML de base
 - 3.3.1. Ossature d'un document HTML (entête, corps, Liens, ...)
 - 3.3.2. Tableaux, Frames, Formulaires
 - 3.3.3. HTML 5.0
 - 3.3.4. Feuilles de style (CSS 3)
 - 3.3.5. JavaScript
 - 3.3.6. Contrôle des formulaires HTML en JavaScript
4. XML
 - 4.1. Structure d'un document XML
 - 4.2. DTD (Document Type Definition)
 - 4.3. XML Schema
 - 4.4. XSLT

Chapitre 3 : Langage de programmation côté serveur (PHP)

1. Introduction
2. Syntaxe de base
 - 2.1. Le passage du HTML au PHP
 - 2.2. Les séparateurs d'Instructions
 - 2.3. Les commentaires

3. Types, variables et opérateurs
4. Structures de contrôles
5. Classes et objets
6. Caractéristiques
 - 6.1. Gestion des erreurs
 - 6.2. Gestion des chargements de fichiers
 - 6.3. Utilisation des fichiers à distance
 - 6.4. Gestion des connexions
 - 6.5. Connexions persistantes aux Bases de Données.
 - 6.6. Gestion des sessions
 - 6.7. Applications Web 3tier en PHP

Chapitre 4 : Services Web : notions de base

1. Introduction
2. Architecture orientée services (SOA)
3. Caractéristiques des services Web
 - 3.1. Définition des services Web
 - 3.2. Architecture des services Web
4. Standards pour les services Web
 - 4.1 SOAP
 - 4.2 WSDL
 - 4.3 UDDI
5. Plateformes de développement des services Web
 - 5.1 Développement des services Web (côté fournisseur)
 - 5.2 Développement des services Web (côté consommateur)
6. Platform .NET et Java.
 - 6.1. JSP
 - 6.2. ASP

Chapitre 5 : Etude de cas : développement d'un service Web (côté fournisseur et puis côté client)

Mode d'évaluation : Examen (60%) , contrôle continu (40%)

Référence :

Cours de développement Web. Disponible sur le site : <https://openclassrooms.com/courses>.

Jean ENGELS. "*PHP 5 – Cours et exercices*". Editions Eyrolles, 2005

Mathieu Lacroix, "*Introduction Web: Cours*". Université paris 13, 2013.

Société Digimind. "*Le Web 2.0 pour la veille et la recherche d'information, Exploitez les ressources du web social*", Digimind, juin 2007.

Semestre : 04

Unité d'enseignement Transversale

Matière : Langue étrangère

Crédits : 2

Coefficient : 1

Objectifs de l'enseignement : *Techniques d'expression écrite et orale en anglais : exposé, soutenance, communication en groupes...*

Connaissances préalables recommandées *Matière d'anglais de L1 et Semestre 3*

Contenu de la matière :

- 4. Techniques d'expression orale en anglais : - Exposés
- Soutenance
- Communication en groupes
- ...

Mode d'évaluation : Examen

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

Mode d'évaluation : Examen (100%)