

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

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

Canevas de mise en conformité

**OFFRE DE FORMATION
L.M.D.**

LICENCE ACADEMIQUE

2025 - 2026

Etablissement	Faculté / Institut	Département

Domaine	Filière	Spécialité
Mathématiques et Informatique	Informatique	Systèmes Informatiques (SI)

الجمهورية الجزائرية الديمقراطية الشعبية

وزارة التعليم العالي والبحث العلمي

نموذج مطابقة

عرض تكوين

ل.م.د

ليسانس أكاديمية

2026-2025

المؤسسة	الكلية / المعهد	القسم

الميدان	الفرع	التخصص
رياضيات وإعلام مالي	إعلام مالي	نظم معلوماتية

**II – Fiche d’organisation semestrielle des enseignements
de la Licence Systèmes Informatiques (SI)**

Semestre 1 :

Unité d’Enseignement	VHS	V.H hebdomadaire				Coeff	Crédits	Mode d'évaluation	
	14 sem	C	TD	TP	Autres			Continu	Examen
UE Fondamental									
UEF11(O/P)		4h30	4h30		6h	6	10		
UEF111 : Analyse 1	84h	3h	3h		3h	4	6	40%	60%
UEF112 : Algèbre 1	42h	1h30	1h30		3h	2	4	40%	60%
UEF12(O/P)		4h30	3h	3h	6h	8	12		
UEF121 : Algorithmique et structure de données 1	105h	3h	1h30	3h	3h	5	7	40%	60%
UEF122 : Structure machine 1	42h	1h30	1h30		3h	3	5	40%	60%
UE Méthodologie									
UEM11(O/P)		1h30			2h	1	2		
UEM111 : Logiciels libres (Open source)	21h	1h30			2h	1	2		100%
UE Transversal									
UET11(O/P)		1h30			2h	1	2		
UET111 : Langue Étrangère	21h	1h30			2h	1	2		100%
UE Découverte									
UED11(O/P)		1h30	1h30		2h	2	4		
UED111:Electricité générale	42h	1h30	1h30		2h	2	4	40%	60%
Total Semestre 1	357h	13h30	9h	3h	18h	18	30		

Semestre 2

Unité d’Enseignement	VHS	V.H hebdomadaire				Coeff	Crédits	Mode d'évaluation	
	14 sem	C	TD	TP	Autres			Continu	Examen
UE Fondamental									
UEF21(O/P)		4h30	3h		6h	6	10		
UEF211 : Analyse 2	63h	3h	1h30		3h	4	6	40%	60%
UEF212 : Algèbre 2	42h	1h30	1h30		3h	2	4	40%	60%
UEF22(O/P)		3h	3h	3h	6h	8	12		
UEF221 : Algorithmique et structure de données 2	84h	1h30	1h30	3h	3h	5	7	40%	60%
UEF222 : Structure machine 2	42h	1h30	1h30		3h	3	5	40%	60%
UE Méthodologie									
UEM21(O/P)		3h	1h30	1h30	4h	2	4		
UEM211 : Logique Mathématique	42h	1h30	1h30		2h	1	2	40%	60%
UEM212 : Introduction à l’intelligence artificielle	42h	1h30		1h30	2h	1	2	40%	60%
UE Découverte									
UED21(O/P)		1h30	1h30		2h	2	4		
UED211 : Electronique fondamentale	42h	1h30	1h30		2h	2	4	40%	60%
Total Semestre 2	357h	12h	9h	4h30	18H	18	30		

Semestre 3

Unité d'Enseignement	VHS	V.H hebdomadaire				Coeff	Crédits	Mode d'évaluation	
	14 sem	C	TD	TP	Autres			Continu	Examen
UE Fondamental									
UEF31(O/P)		3h	3h	3h	6h	7	11		
UEF311:Architecture des ordinateurs	63h	1h30	1h30	1h30	3h	3	5	40%	60%
UEF312:Algorithmique et structure de données 3	63h	1h30	1h30	1h30	3h	4	6	40%	60%
UEF32(O/P)		3h	1h30	3h	6h	6	9		
UEF321:Systèmes d'information	42h	1h30	1h30		3h	2	4	40%	60%
UEF322:Programmation orienté objet 1	63h	1h30		3h	3h	4	5	40%	60%
UE Méthodologie									
UEM31 (O/P)		3h	1h30	1h30	4h	4	8		
UEM311:Méthodes Numériques	42h	1h30		1h30	2h	2	4	40%	60%
UEM312:Probabilités et statistique 1	42h	1h30	1h30		2h	2	4	40%	60%
UE Transversal									
UET31 (O/P)		1h30		1h30	2h	1	2		
UET311:Gestion de projets	42h	1h30		1h30	2h	1	2	40%	60%
Total Semestre 3	357h	10h30	6h	9h	18h	18	30		

Semestre 4

Unité d'Enseignement	VHS	V.H hebdomadaire				Coeff	Crédits	Mode d'évaluation	
	14 sem	C	TD	TP	Autres			Continu	Examen
UE Fondamental									
UEF41(O/P)		4h30	3h	1h30	6h	7	10		
UEF411: Système d'exploitation 1	84h	3h	1h30	1h30	3h	4	5	40%	60%
UEF422: Théorie des langages	42h	1h30	1h30		3h	3	5	40%	60%
UEF42(O/P)		3h	1h30	4h30	6h	6	10		
UEF421: Base de données	63h	1h30	1h30	1h30	3h	3	5	40%	60%
UEF422: Programmation orienté objet 2	63h	1h30		3h	3h	4	5	40%	60%
UE Méthodologie									
UEM41 (O/P)		3h	3h		06h	4	8		
UEM411: Programmation Linéaire	42h	1h30	1h30		3h	2	4	40%	60%
UEM412: Probabilités et Statistique 2	42h	1h30	1h30		3h	2	4	40%	60%
UE Transversal									
UET41 (O/P)		1h30			2h	1	2		
UED411: Entreprenariat	21h	1h30			299h	1	2		100%
Total Semestre 4	357h	12h	7h30	6h	20h	18	30		

Semestre 5

Unité d’Enseignement	VHS	V.H hebdomadaire				Coeff	Crédits	Mode d'évaluation	
	14 sem	C	TD	TP	Autres			Continu	Examen
UE Fondamental									
UEF51(O/P)		3h	3h	3h	6h	7	10		
UEF511: Réseaux	63h	1h30	1h30	1h30	3h	4	5	40%	60%
UEF512:Compilation	63h	1h30	1h30	1h30	3h	3	5	40%	60%
UEF52(O/P)		3h	1h30	3h	6h	6	10		
UEF521:Génie Logiciel	42h	1h30	1h30		3h	3	5	40%	60%
UEF522:Développement d’Applications Web	63.33h	1h30		3h	3h	3	5	40%	60%
UE Méthodologie									
UEM51 (O/P)		3h	1h30	1h30	6h	4	8		
UEM511: Outils de programmation scientifique	42h	1h30		1h30	3h	2	4	40%	60%
UEM512:Théorie des graphes	42h	1h30	1h30		3h	2	4	40%	60%
UE Transversal									
UET51 (O/P)		1h30		1h30	2h	1	2		
UET511:Rédaction Scientifique	42h	1h30		1h30	2h	1	2	40%	60%
Total Semestre 5	357h	10h30	6h	9h	20h	18	30		

Semestre 6

Unité d’Enseignement	VHS	V.H hebdomadaire				Coeff	Crédits	Mode d’évaluation	
	14 sem	C	TD	TP	Autres			Continu	Examen
UE Fondamental									
UEF61 (O/P)		3h	1h30	3h00	6h	6	10		
UEF611: Système d'exploitation 2	63h	1h30	1h30	1h30	3h	3	5	40%	60%
UEF622:Sécurité Informatique	42h	1h30		1h30	3h	3	5	40%	60%
UEF62 (O/P)		3h	1h30	1h30	6h	6	10		
UEF621:Recherche d’information	42h	1h30	1h30		3h	3	5	40%	60%
UEF622: Fondements de l’Intelligence Artificielle	42h	1h30		1h30	3h	3	5	40%	60%
UE Méthodologie									
UEM61 (O/P)		1h30			12h	5	8		
UEM611:Projet					10h	4	6		100%
UEM612:Intelligence métier	21h	1h30			2h	1	2		100%
UE Transversal									
UET61 (O/P)		1h30			2h	1	2		
UET611:Déontologie de l’informatique	21h	1h30			2	1	2		100%
Total Semestre 6	231h	9h	3h00	4h30	26h	18	30		

Récapitulatif global de la formation :(indiquer le VH global séparé en cours, TD, TP... pour les 06 semestres d'enseignement, pour les différents types d'UE)

UE VH	UEF	UEM	UED	UET	Total
Cours	568h30	229h30	43h30	84h	925h30
TD	469	85h30	22h30	19h30	596h30
TP	333h30	102h		42h	477h30
Travail personnel					
Autre (préciser)	18h	474h	56h	107h	655
Total	1389	891	122	252h30	2654h30
Crédits	124	37	10	10	
% en crédits pour chaque UE	68%	21%	5.5%	5.5%	100%

III - Programme détaillé par matière des semestres
(1 fiche détaillée par matière)

(Tous les champs sont à renseigner obligatoirement)

Semestre:1

Unité d'enseignement : UEF11

Matière : Analyse 1

Crédits : 6

Coefficient : 4

Objectif du cours:L'objectif de ce module est de familiariser les étudiants avec le vocabulaire ensembliste, de donner des différentes méthodes de convergence des suites réelles et les différents aspects de l'analyse des fonctions d'une variable réelle.

Connaissances préalables recommandées : Niveau terminale.

Contenu de la matière :

Chapitre 1 : Le Corps des Réels

$\mathbb{R}$ est un corps commutatif, $\mathbb{R}$ est un corps totalement ordonné, Raisonnement par récurrence, $\mathbb{R}$ est un corps valué, Intervalles, Bornes supérieure et inférieure d'un sous ensemble de $\mathbb{R}$, $\mathbb{R}$ est un corps archimédien, Caractérisation des bornes supérieure et inférieure, La fonction partie entière, Ensembles bornés, Prolongement de $\mathbb{R}$: Droite numérique achevée $\mathbb{R}$, Propriétés topologiques de $\mathbb{R}$, Parties ouvertes fermées.

Chapitre 2 : Le Corps des Nombres Complexes

Opérations algébriques sur les nombres complexes, Module d'un nombre complexe z , Représentation géométrique d'un nombre complexe, Forme trigonométrique d'un nombre complexe, Formules d'Euler, Forme exponentielle d'un nombre complexe, Racines n -ième d'un nombre complexe.

Chapitre 3 : Suites de Nombres réels

Suites bornées, suites convergentes, Propriétés des suites convergentes, Opérations arithmétiques sur les suites convergentes, Extensions aux limites infinies, Infiniment petit et infiniment grand, Suites monotones, Suites extraites, Suite de Cauchy, Généralisation de notion de la limite, Limite supérieure, Limite inférieure, Suites récurrentes.

Chapitre 4 : Fonctions réelles d'une variable réelle

Graphes d'une fonction réelle d'une variable réelle, Fonctions paires-impaires, Fonctions périodiques, Fonctions bornées, Fonctions monotones, Maximum local, Minimum local, Limite d'une fonction, Théorèmes sur les limites, Opérations sur les limites, Fonctions continues, Discontinuités de première et de seconde espèce, Continuité uniforme, Théorèmes sur les fonctions continues sur un intervalle fermé, Fonction réciproque continue, Ordre d'une variable-équivalence (Notation de Landau).

Chapitre 5: Fonctions dérivables

Dérivée à droite, dérivée à gauche, Interprétation géométrique de la dérivée, Opérations sur les fonctions dérivables, Différentielle-Fonctions différentiables, Théorème de Fermat, Théorème de Rolle, Théorème des accroissements finis, Dérivées d'ordre supérieur, Formule de Taylor, Extrémum local d'une fonction, Bornes d'une fonction sur un intervalle, Convexité d'une courbe. Point d'inflexion, Asymptote d'une courbe, Construction du graphe d'une fonction.

Chapitre 6 : Fonctions Élémentaires

Logarithme népérien, Exponentielle népérienne, Logarithme de base quelconque, Fonction puissance, Fonctions hyperboliques, Fonctions hyperboliques réciproques.

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

Références

1. J.-M. Monier, Analyse PCSI-PTSI, Dunod, Paris 23.
2. Y. Bougrov et S. Nikolski, Cours de Mathématiques Supérieures, Editions Mir, Moscou, 1983.
3. N. Piskounov, Calcul différentiel et intégral, Tome 1, Editions Mir, Moscou, 1980.
4. K. Allab, Eléments d'Analyse, OPU, Alger, 1984.
5. B. Calvo, J. Doyen, A. Calvo, F. Boschet, Cours d'analyse, Librairie Armand Colin, Paris, 1976.
6. J. Lelong-Ferrand et J. M. Arnaudiès, Cours de mathématiques, tome 2, Edition Dunod, 1978.

Semestre:1

Unité d'enseignement : UEF11

Matière : Algèbre 1

Crédits : 4

Coefficient : 2

Objectifs de l'enseignement : Ce module permet d'introduire les notions de base de l'algèbre et de la théorie des ensembles.

Connaissances préalables recommandées : Notions d'algèbre classique

Contenu de la matière :

Chapitre 1 : Notions de logiques.

- Tables de vérité, quantificateurs et types de raisonnement (direct, par l'absurde, par contraposée et par récurrence).

Chapitre 2 : Ensembles et applications.

- Définitions et exemples.
- Applications : injection, surjection, bijection, image directe, image réciproque, restriction et prolongement.

Chapitre 3 : Relations binaires sur un ensemble.

- Définitions de base : relation réflexive, symétrique, antisymétrique, transitive.
- Relation d'ordre- Définition. Ordre total et partiel.
- Relation d'équivalence : classe d'équivalence.

Chapitre 4 : Structures algébriques.

- Loi de composition interne. Partie stable. Propriétés d'une loi de composition interne.
- Groupes-Définitions. Sous-groupe-Exemples-Homomorphisme de groupes- isomorphisme de groupes. Donner des exemples de groupes finis $\mathbb{Z}/n\mathbb{Z}$ ($n= 1, 2, 3, \dots$) et le groupe de permutations S_3 .
- Anneaux-Définition- Sous anneaux. Règles de calculs dans un anneau. Eléments inversibles, diviseurs de zéro-Homomorphisme d'anneaux-Idéaux.
- Corps-Définitions-Traiter le cas d'un corps fini à travers l'exemple $\mathbb{Z}/p\mathbb{Z}$ ou p est premier, $\mathbb{R}$ et $\mathbb{C}$

Chapitre 5 : Anneaux de polynômes.

- Polynôme, degré.
- Construction de l'anneau des polynômes.
- Arithmétique des polynômes-Divisibilité-Division euclidienne-Pgcd et ppcm de deux polynômes- Polynômes premiers entre eux-Décomposition en produit de facteurs irréductibles.
- Racines d'un polynôme-Racines et degré -Multiplicité des racines.

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

Références

1. M. Mignotte et J. Nervi, Algèbre : licences sciences 1ère année, Ellipses, Paris, 24.
2. J. Franchiniet J. C. Jacquens, Algèbre : cours, exercices corrigés, travaux dirigés, Ellipses, Paris, 1996.
3. C. Degrave et D. Degrave, Algèbre 1ère année : cours, méthodes, exercices résolus, Bréal, 23.
4. S. Balac et F. Sturm, Algèbre et analyse : cours de mathématiques de première année avec exercices corrigés, Presses Polytechniques et Universitaires, 23.

Semestre:1

Unité d'enseignement: UEF12

Matière : Algorithmique et structure de données 1

Crédits : 7

Coefficient : 5

Objectifs de l'enseignement : Un algorithme est une démarche méthodologique de résolution d'un problème étape par étape. Il définit une suite d'instructions à exécuter dans un certain ordre pour obtenir le résultat souhaité. Un algorithme est généralement créé indépendamment du langage de programmation, c'est-à-dire qu'un algorithme peut être implémenté dans plusieurs langages de programmation. Une structure des données est une manière particulière de stocker et d'organiser les informations dans la mémoire d'un ordinateur afin qu'elles puissent être récupérées et utilisées de manière efficace.

L'objectif de ce module est d'introduire et de construire les bases de la conception d'algorithmes et de structures de données pour la résolution de problèmes. Le « langage C » est recommandé comme support d'implémentation des algorithmes.

Connaissances préalables recommandées : Connaissances de base en mathématiques.

Contenu de la matière :

Chapitre 1 : Introduction

1. Bref historique de l'informatique
2. La structure fonctionnelle d'un ordinateur (mémoire, processeur, E/S)
3. Le matériel (hardware) et le logiciel (software)
4. Langage de programmation
5. Cycle de résolution d'un problème par ordinateur (Analyse, Algorithme, Programme, Exécution)

Chapitre 2 : Algorithme séquentiel simple

1. Exemples simples d'algorithmes de la vie courante
2. Définition d'un algorithme
3. Structuration d'un algorithme (titre de l'algorithme, déclarations, instructions, indentation)
4. Notion de variable et de constante algorithmique
5. Les types de bases (entier réel, caractère)
6. Etapes simples : lecture, affichage, affectation
7. Résolution algorithmique de problèmes simples
8. Représentation graphique d'un algorithme : l'organigramme
9. Traduction d'un algorithme simple en langage C

Chapitre 3 : Les structures conditionnelles

1. L'importance de prise de décision lors de la résolution d'un problème
2. Structure conditionnelle simple (Si ... Alors FinSi)
3. Structure conditionnelle double (Si ... Alors ... Sinon FinSi)
4. Imbrication de structures conditionnelles
5. Structure conditionnelle de choix multiples
6. Instructions conditionnelles en langage C

Chapitre 4 : Les structures répétitives

1. Importance de la répétition dans la résolution de problèmes (exemples)
2. Notion de compteur de boucle et de condition de répétition (ou d'arrêt)
3. La boucle (TantQue FinTQ)
4. La boucle (Répéter Jusqu'à)
5. La boucle (Pour FinPour)
6. Imbrication de boucles
7. Les structures conditionnelles du langage C

Chapitre 5 : Les tableaux et les chaînes de caractères

1. Introduction : nécessité de données tabulaires (exemples)
2. Le type tableau à une dimension (1D)
3. Notion d'indice, accès à un élément du tableau, manipulation d'un tableau
4. Le tableau à deux dimensions (2D), ou matrice
5. Quelques applications des tableaux 1D et 2D
6. Le type chaîne de caractères (cas particulier de tableau 1D)
7. Quelques applications des chaînes de caractères
8. Les tableaux 1D, 2D et les chaînes de caractères en langage C

Chapitre 6 : Les types personnalisés

1. Introduction : nécessité de définir des types personnalisée (exemples)
2. Le type structure (enregistrement) et son utilisation

3. Les énumérations
4. Application en langage C (struct, enum, union, typedef)

Recommandations :

- La séance de TP est utilisée pour introduire le langage C au fur et à mesure en utilisant un IDE tel que CodeBlocks ou DevCpp
- Comme il s'agit d'un premier module d'algorithmique, les fiches de TD doivent comporter un nombre important d'exercices. Il faut encourager les étudiants à préparer les exercices et à traduire tous les algorithmes en programmes.
- Insister sur la présentation d'un algorithme et d'un programme (choix des variables, indentation, commentaires,)

Mode d'évaluation : Examen (60%), Contrôle continu (en TD et test de TP) (40%)

Références

1. T. H. Cormen, C. Leiserson, R. Rivest & C. Stein. **Introduction to Algorithms**. The MIT Press, 4th Edition, 2022.[Disponible au format PDF](#)
2. T. H. Cormen. **Algorithmes, notions de base**. Collection Sciences SUP, Dunod, 2013.
3. S. Oualline. **Practical C Programming**. O'REILLY, 3rd Edition.[Disponible au format PDF](#)
4. Canteaut. **Programmation en Langage C**. INRIA projets CODES.[Disponible au format PDF](#)

Semestre:1

Unité d'enseignement : UEF12

Matière : Structure machine 1

Crédits : 5

Coefficient : 3

Objectifs de l'enseignement :

Un premier but de cette matière est de présenter et d'approfondir les notions concernant les différents systèmes de numération ainsi que la représentation de l'information qu'elle soit de type numérique ou caractère. Les bases de l'algèbre de Boole sont, elles aussi, abordées de façon approfondie.

Ces connaissances vont servir de plateforme pour d'autres aspects en relation avec l'ordinateur (architecture des ordinateurs, programmation, systèmes d'exploitation, réseaux, ...).

Connaissances préalables recommandées :

Mathématiques élémentaires.

Contenu de la matière :

Chapitre 1 :

- Introduction générale.

Chapitre 2 : Les systèmes de numération

- Définition
- Présentation des systèmes décimal, binaire, octal et hexadécimal.
- Conversion entre ces différents systèmes.
- Opérations de base dans le système binaire :
 - Addition
 - Soustraction
 - Multiplication
 - Division

Chapitre 3 : La représentation de l'information

- Le codage binaire :
 - Le codage binaire pur.
 - Le code binaire réfléchi (ou code DE GRAY)
 - Le code DCB (Décimal codé binaire)
 - Le code excède de trois.
- Représentation des caractères :
 - Code EBCDIC
 - Code ASCII
 - Code UTF.
- Représentation des nombres :
 - 1- Nombres entiers :
 - Représentation non signée.
 - Représentation avec signe et valeur absolue.
 - Complément à 1 (ou Complément restreint)
 - Complément à 2 (ou Complément Vrai)
 - 2- Les nombres fractionnaires :
 - Virgule fixe.
 - Virgule flottante (norme IEEE 754)

Chapitre 4 : L'algèbre de Boole binaire

- Définition et axiomes de l'algèbre de Boole.
- Théorèmes et propriétés de l'algèbre de Boole.
- Les opérateurs de base :
 - ET, OU, négation logique.
 - Représentation schématique.
- Autres opérateurs logiques :
 - Circuits NAND et NOR
 - Ou exclusif.
 - Implication.
 - Représentation schématique.
- Table de vérité.
- Expressions et fonctions logiques.
- Ecriture algébrique d'une fonction sous première et deuxième forme normale
- Expression d'une fonction logique avec des circuits NANDs ou NOR exclusivement.

- Schéma logique d'une fonction.
- Simplification d'une fonction logique :
 - Méthode algébrique.
 - Tableaux de Karnaugh.
 - Méthode de quine-mc cluskey.

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

Références

1. John R. Gregg, Ones and Zeros: Understanding Boolean Algebra, Digital Circuits, and the Logic of Sets 1st Edition , Wiley & sons Inc. publishing, 1998, ISBN: 978-0-7803-3426-7.
2. Bradford Henry Arnold , Logic and Boolean Algebra, Dover publication, Inc., Mineola, New York, 2011, ISBN-13: 978-0-486-48385-6
3. Alain Cazes, Joëlle Delacroix, Architecture Des Machines Et Des Systèmes Informatiques : Cours et exercices corrigés, 3^e édition, Dunod 28.
4. Donald D. Givone :«Digital Principles and Design», Mc GrawHill, 23

Semestre:1

Unité d'enseignement : UEM11

Matière : Logiciels libres (Open source)

Crédits : 2

Coefficient : 1

Contenu de la matière :

Objectifs de l'enseignement : Familiarisation avec l'outil informatique et l'Internet.

Connaissances préalables recommandées : Connaissances générales en informatique.

Contenu de la matière :

Chapitre 1 : Technologies de l'information

1.1. Définitions

1.2. Outils

1.2.1. Matériels (Hardware)

1. Ordinateurs : PC de bureau, ordinateurs portables, serveurs, stations de travail.
2. Réseaux de communication : routeurs, commutateurs (switch), modems, fibres optiques, Wi-Fi.
3. Périphériques : imprimantes, scanners, claviers, souris, caméras, capteurs.
4. Dispositifs mobiles : smartphones, tablettes, assistants numériques personnels (PDA).
5. Puces intelligentes : RFID, NFC, processeurs embarqués, capteurs IoT (Internet of Things).

1.2.2. Logiciels (Software)

1. Systèmes d'exploitation : Windows, Linux, macOS, Android, iOS.
2. Applications métiers : ERP (Enterprise Resource Planning), CRM (Customer Relationship Management), logiciels de gestion des ressources humaines, de comptabilité, etc.
3. Logiciels de bureautique : traitement de texte (Word), tableurs (Excel), présentations (PowerPoint), outils de collaboration (Google Workspace, Microsoft 365).

1.3. Applications

1. Espaces de communications : Internet, Intranet, Extranet
2. Multimédia : Audioconférence, visioconférence
3. Echange de données informatisées (EDI)
4. Workflows

Chapitre 2 : Outils Open source

1. Introduction (historique, avantages / inconvénients et licence)
2. Environnement de développement (Introduction à Linux, Introduction aux éditeurs de code)
3. Bureautique (Suite bureautique LibreOffice)
4. Collaboration (Stockage et partage)
5. Contribuer à un projet open source

Mode d'évaluation : Examen (100 %)

Références

1. Collectif Eni, Microsoft Office 2016 Word, Excel, PowerPoint, Outlook 2016 - Fonctions de base, Eni Collection : Référence bureautique, 2016
2. Dan Gookin, Greg Harvey, Word et Excel 2016 pour les nuls, First, Collection : Pour les nuls - Poche (informatique), 2016
3. Myriam GRIS, Initiation à Internet, Eni editions, 29

Semestre:1

Unité d'enseignement : UET11

Matière : Langue étrangère

Crédits : 2

Coefficient : 1

Objectifs de l'enseignement :

Le but de cette matière est de permettre aux étudiants d'améliorer leurs compétences linguistiques générales sur le plan de la compréhension et de l'expression, ainsi que l'acquisition du vocabulaire spécialisé de l'anglais scientifique et technique.

Connaissances préalables recommandées : Connaissances élémentaires en Anglais

Contenu de la matière :

1. Rappels sur les bases essentielles de la grammaire anglaise

- Les temps (présent, passé, futur,...)
- Les verbes : réguliers et irréguliers.
- Les adjectifs.
- Les auxiliaires.
- Construire des phrases en anglais : affirmatives, négatives et interrogatives, Formation des phrases.
- Autres structures de la grammaire anglaise.

2. Vocabulaire, expressions et construction de textes techniques

- L'informatique et internet : vocabulaire technique.
- Construction de textes techniques en anglais.

Mode d'évaluation : Examen (100 %)

Références

1. Murphy. English Grammar in Use. Cambridge University Press. 3rd edition, 24
2. M. Mc Carthy et F. O'Dell, English vocabulary in use, Cambridge University Press, 1994
3. L. Rozakis, English grammar for the utterly confused, Mc Graw-Hill, 1st edition, 23
4. Oxford Progressive English books.

Semestre:1

Unité d'enseignement : UED11

Matière : Electricité générale

Crédits : 4

Coefficient : 2

Objectifs de l'enseignement :

A la fin de ce cours, l'étudiant devra acquérir les connaissances élémentaires en électricité et magnétisme (Calcul des champs et Potentiels électrique et magnétique, Calcul des courants,...), de façon à pouvoir analyser et interpréter les phénomènes qui y sont reliés.

Connaissances préalables recommandées : Notions élémentaires de Physique

Contenu de la matière :

Chapitre 1 : Electrostatique

- Forces électrostatiques
- Champs
- Potentiel
- Dipôle électrique
- Théorème de Gauss

Chapitre 2 : Les conducteurs

- Influence totale et partielle
- Calcul des capacités – Resistances – Lois
- Loi d'ohm généralisée

Chapitre 3 : Electrocinétique

- Loi d'Ohm
- Loi de Kirchhoff
- Loi de Thévenin - Norton

Chapitre 4 : Magnétostatique

- Force magnétostatique (Lorentz et Laplace)
- Champs magnétiques
- Loi de Biot et Savark

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

Références

1. T. Neffati. Electricité générale. 28. Editions Dunod
2. D. Bohn. . Electricité générale. 29. Editions SAEP
3. Y. Granjon. Electricité générale. 29. Editions Dunod

Semestre:2

Unité d'enseignement : UEF21

Matière : Analyse 2

Crédits : 6

Coefficient : 4

Objectifs de l'enseignement :

- Présenter les applications des développements limités : utilisation des DL pour simplifier des expressions complexes dans les algorithmes et évaluer les erreurs dans les calculs numériques et les approximations.
- Présenter aux étudiants les différents aspects du calcul intégral : intégrale de Riemann, différentes techniques de calcul des primitives.
- Initiation à la résolution des équations différentielles de premier et deuxième ordre.
- Généraliser la notion de continuité et de dérivabilité à des fonctions définies sur un ensemble de vecteurs ($\mathbb{R}^n$)

Connaissances préalables recommandées : Analyse Mathématique 1, Algèbre 2 (espaces vectoriels)

Chapitre 1 : Développements limités

Développements limités polynomiaux (D.L), et opérations sur les D.L.

Généralisation des développements limités.

Application au calcul de limites et à l'étude des branches infinies.

Chapitre 2 : Intégrales indéfinies

Intégrale indéfinie, Quelques propriétés de l'intégrale indéfinie, Méthodes d'intégration, Intégration par changement de variable, Intégration par parties, Intégration d'expressions rationnelles, Intégration de fonctions irrationnelles.

Chapitre 3 : Intégrales définies

Intégrale de Riemann, Intégrale définie, Propriétés des intégrales définies, Intégrale fonction de sa borne supérieure, Formule de Newton-Leibniz, Inégalité Cauchy-Schwarz, Sommes de Darboux- Conditions de l'existence de l'intégrale, Propriétés des sommes de Darboux, Intégrabilité des fonctions continues et monotones.

Chapitre 4 : Équations différentielles du premier ordre

Généralités, Classification des équations différentielles du premier ordre, Équation à variables séparables, Équations homogènes, Équations linéaires, Méthode de Bernoulli, Méthode de la variation de la constante de Lagrange, Équation de Bernoulli, Équation différentielle totale, Équation de Riccati.

Chapitre 5 : Équations différentielles du second ordre à coefficients constants

Équations différentielles du second ordre homogènes à coefficients constants,

Équations différentielles du second ordre non homogènes à coefficients constants,

Méthodes de résolutions des équations différentielles du second ordre à coefficients constants.

Chapitre 6 : Fonctions de plusieurs variables

Généralités, normes sur $\mathbb{R}^n$ (euclidienne, d'ordre p , infinie), distance induite par une norme, propriétés.

Boules ouvertes, fermées, sphères sur un espace vectoriel normé.

Dérivées partielles, gradient, des fonctions de deux variables.

Différentiabilité, continuité, accroissement finis.

Points critiques : maximum et minimum locaux.

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

Recommandation : Pour une meilleure compréhension du chapitre 6, il est recommandé de traiter des exemples dans IR2.

Références

1. K. Allab, Eléments d'Analyse, OPU, Alger, 1984.
2. J.-M. Monier, Analyse PCSI-PTSI, Dunod, Paris 23.
3. Y. Bougrov et S. Nikolski, Cours de Mathématiques Supérieures, Editions Mir, Moscou, 1983.
4. N. Piskounov, Calcul différentiel et intégral, Tome 1, Editions Mir, Moscou, 1980.
5. Robert G. Bartle, Donald R. Sherbert, Introduction to Real Analysis (4th Edition), 2011
6. Jiri Lebel, Basic Analysis II; Introduction to Real Analysis, Volume II (2023) (6th Edition)
7. G. B. Thomas, M. D. Weir, and Joel Hass, Calculus and Analytic Geometry, Pearson 2014 (13ème édition)

Semestre: 2

Unité d'enseignement : UEF21

Matière : Algèbre 2

Crédits : 4

Coefficient : 2

Objectifs de l'enseignement :

Etude des concepts fondamentaux relatifs aux espaces vectoriels de dimension finie.

Assimiler les bases de calcul matriciel afin d'acquérir des connaissances suffisantes pour aborder l'algèbre 3.

Connaissances préalables recommandées : Algèbre 1.

Contenu de la matière :

Chapitre 1 : Espaces vectoriels.

- Définition d'un espace vectoriel. Sous espace vectoriel. Exemples.
- Familles libres. Génératrices. Bases. Dimension.
- Espace vectoriel de dimension finie (propriétés). Sous espace vectoriel supplémentaire.

Chapitre 2 : Applications linéaires.

- Définition.
- Image et noyau d'une application linéaire.
- Propriétés d'une application linéaire en dimension finie
- Rang d'une application, théorème du rang.

Chapitre 3 : Matrices et déterminants.

- Définition. Matrices particulières
- Opérations sur les matrices (somme, produit de deux matrices.)
- Matrice associée à une application linéaire et propriétés
- Espace vectoriel des matrices.
- Anneau des matrices carrées, matrice inversible.
- Matrice de changement de base (matrice de passage).
- Matrices semblables et matrices équivalentes
- Déterminant d'une matrice carrée et propriétés.
- Méthodes de calcul de l'inverse d'une matrice carrée inversible (méthode de Gauss et des cofacteurs).

Chapitre 4 : Résolution de systèmes d'équations.

- Système d'équations – écriture matricielle - rang d'un système d'équations.
- Résolution d'un système linéaire à n équations et m inconnues.
- Méthode de Cramer. Théorème de Rouché-Frobenius (système linéaire général).

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

Recommandation : Pour rapprocher les notions d'algèbre linéaire à l'informatique, il est recommandé de traiter des exemples et des exercices d'application à l'informatique.

Références

1. J. Franchini et J. C. Jacquens, Algèbre : cours, exercices corrigés, travaux dirigés, Ellipses, Paris, 2014.
2. Lieven Vandenberghe et Stephen P. Boyd. Introduction to applied linear algebra: vectors, matrices and least squares. 2018
3. Sheldon Axler. Linear Algebra Done Right" (3rd Edition). Springer. 2015.
4. David C. Lay, Steven R. Lay, et Judi J. McDonald. Linear Algebra and Its Applications (5e édition, 2016)
Ce livre aborde les concepts d'algèbre linéaire avec un accent particulier sur les applications, y compris celles en informatique et en sciences des données.
5. Gene H. Golub et Charles F. Van Loan. Matrix Computations (4e édition, 2013)
Bien que plus technique, cet ouvrage est fondamental pour comprendre les algorithmes de calcul matriciel, essentiels en informatique, en traitement d'images et en apprentissage automatique.
6. John L. Weatherwax. Linear Algebra and Its Applications in Computer Science. 2014.
Ce livre se concentre spécifiquement sur les applications de l'algèbre linéaire dans divers domaines de l'informatique, y compris la théorie des graphes et l'apprentissage automatique.
7. J. J. Sylvester. Linear Algebra for Computer Science (2020)
Ce livre propose une introduction à l'algèbre linéaire avec des exemples et des exercices orientés vers des applications en informatique.

Semestre:2

Unité d'enseignement : UEF22

Matière : Algorithmique et structure de données 2

Crédits : 7

Coefficient : 5

Objectifs de l'enseignement : permettre à l'étudiant d'acquérir des notions fondamentales de la programmation

Connaissances préalables recommandées : Notions élémentaires de mathématiques

Contenu de la matière :

Chapitre 1 : Les sous-programmes : Fonctions et Procédures

1. Introduction
2. Définitions
3. Les variables locales et les variables globales
4. Le passage des paramètres
5. La récursivité

Chapitre 2 : Les fichiers

1. Introduction
2. Définition
3. Types de fichier
4. Manipulation des fichiers

Chapitre 3 : Les listes chaînées

1. Introduction
2. Les pointeurs
3. Gestion dynamique de la mémoire
4. Les listes chaînées
5. Opérations sur les listes chaînées
6. Les listes doublement chaînées
7. Les listes chaînées particulières
 - 7.1. Les piles
 - 7.2. Les files

NB : TPs en C (Complémentaires aux TDs).

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

Références

1. Thomas H. Cormen, Algorithmes Notions de base *Collection : Sciences Sup, Dunod*, 2013.
2. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest Algorithmique - 3ème édition - Cours avec 957 exercices et 158 problèmes Broché, Dunod, 2010.
3. Rémy Malgouyres, Rita Zrour et Fabien Feschet. *Initiation à l'algorithmique et à la programmation en C : cours avec 129 exercices corrigés*. 2^{ième} Edition. Dunod, Paris, 2011. ISBN : 978-2-10-055703-5.
4. Damien Berthet et Vincent Labatut. *Algorithmique & programmation en langage C - vol.1 : Supports de cours*. Licence. Algorithmique et Programmation, Istanbul, Turquie. 2014, pp.232.
5. Damien Berthet et Vincent Labatut. *Algorithmique & programmation en langage C - vol.2 : Sujets de travaux pratiques*. Licence. Algorithmique et Programmation, Istanbul, Turquie. 2014, pp.258. <cel-01176120>
6. Damien Berthet et Vincent Labatut. *Algorithmique & programmation en langage C - vol.3 : Corrigés de travaux pratiques*. Licence. Algorithmique et Programmation, Istanbul, Turquie. 2014, pp.217. <cel-01176121>
7. Claude Delannoy. *Apprendre à programmer en Turbo C*. Chihab- EYROLLES, 1994.

Semestre:2

Unité d'enseignement : UEF22

Matière : Structure Machine 2

Crédits : 5

Coefficient : 3

Objectifs de l'enseignement : A la fin du semestre, les étudiants bénéficient de connaissances de base sur l'architecture des ordinateurs et le principe de fonctionnement de chacun des composants. Ces connaissances vont servir de plateforme pour d'autres aspects en relation avec l'ordinateur (programmation, base de données, réseaux,...).

Connaissances préalables recommandées : Les étudiants doivent avoir des notions élémentaires en informatique.

Contenu de la matière :

Chapitre 1 : Introduction

Chapitre 2 : La logique combinatoire

- Définition.
- Les circuits combinatoires.
- Étapes de conception d'un circuit combinatoire :
 - Etablissement de la table de vérité.
 - Simplification des fonctions logiques.
 - Réalisation du schéma logique.
- Etude de quelques circuits combinatoires usuels :
 - Le demi-additionneur.
 - L'additionneur complet.
 - L'additionneur soustracteur (en complément vrai)
 - Les décodeurs.
 - Les multiplexeurs.
 - Les encodeurs de priorité.
 - Les démultiplexeurs.
- Autres exemples de circuits combinatoires.

Chapitre 3 : La logique séquentielle.

- Définition.
- Les bascules (RS, JK, D)
- Les registres (à chargement parallèle et à décalage)
- Les mémoires.
- Synthèse d'un circuit séquentiel (automates):
 - Automate de Moore et automate de Mealy.
 - Graphe et matrice de transition.
 - Choix des bascules et codage des états.
 - Matrice d'excitation des bascules.
 - Simplification des fonctions logiques.
 - Etablissement du schéma logique.
- Réalisation d'automates :
 - Les compteurs/décompteurs.
 - Autres exemples d'automates.

Chapitre 4 : Les circuits intégrés.

- Définition
- Etude des caractéristiques d'un circuit intégré simple (exemple circuit ou 7432)
- Notions sur la réalisation du montage d'un circuit combinatoire simple en utilisant des circuits intégrés.

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

Références

1. John R. Gregg, Ones and Zeros: Understanding Boolean Algebra, Digital Circuits, and the Logic of Sets 1st Edition , Wiley & sons Inc. publishing, 1998, ISBN: 978-0-7803-3426-7.
2. Bradford Henry Arnold , Logic and Boolean Algebra, Dover publication, Inc., Mineola, New York, 2011, ISBN-13: 978-0-486-48385-6
3. Alain Cazes, Joëlle Delacroix, architecture des machines et des systèmes informatiques : Cours et exercices corrigés, 3^e édition, Dunod 28.

Semestre:2

Unité d'enseignement : UEM21

Matière : Logique mathématique

Crédits : 4

Coefficient : 2

Objectifs de l'enseignement : Formalisation du raisonnement humain

Connaissances préalables recommandées : connaissances de base en mathématiques et en algèbre de Boole.

Contenu de la matière :

Chapitre 1 : Introduction

1. Objets de la logique
2. Syntaxe et sémantique

Chapitre 2 : Logique des propositions

1. Syntaxe
 - a. Les propositions
 - b. Les connecteurs logiques
 - c. Variables et formules propositionnelles
 - d. Substitution dans une formule
 - e. Formules logiques et arbres
2. Sémantique
 - a. Interprétation
 - b. Tables de vérité
 - c. Tautologies et antilogies
 - d. Equivalence sémantique
 - e. Formes normales conjonctives et disjonctives
 - f. Satisfaisabilité et validité
3. Résolution
 - a. Réfutation
 - b. Mise en forme clausale
 - c. Règle de résolution propositionnelle
 - d. La méthode de résolution propositionnelle

Chapitre 3: Logique des prédicats

1. Syntaxe
 - a. Termes
 - b. Prédicats
 - c. Quantificateurs
 - d. Formules
 - i. Portée d'un identificateur
 - ii. Variables libres, variables liées
2. Sémantique
 - a. Structure
 - b. 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, 1967.
- R. Cori. Logique mathématique. Tome 1 : Calcul propositionnel, Algèbre de Boole, calcul des prédicats. Dunod, 23.

Semestre:2

Unité d'enseignement : UEM21

Matière : Introduction à l'intelligence artificielle

Crédits : 2

Coefficient : 1

Objectifs de l'enseignement :

Ce module a pour vocation de démystifier l'IA, stimuler la curiosité et initier à son utilisation concrète et propre dans un cadre académique. Il vise à offrir une découverte accessible, concrète et stimulante de l'IA, en combinant des apports théoriques et des activités pratiques orientées vers les usages académiques.

À la fin du cours, l'étudiant sera capable de :

- Comprendre les principes fondamentaux de l'intelligence artificielle.
- Identifier les domaines d'application de l'IA dans la vie courante et dans les études universitaires.
- Discuter les enjeux éthiques, sociaux et économiques de l'IA.
- Utiliser quelques outils pratiques d'IA pour améliorer les activités scientifiques.

Contenu de la matière :

Chapitre 1 : Définitions de l'IA ?

- Définition simple et historique
- Différences entre IA faible, forte, symbolique et connexionniste
- Exemples concrets d'IA dans le quotidien
- Domaines de l'IA
 1. Apprentissage automatique, traitement du langage naturel, vision par ordinateur, robots
 2. Applications : médecine, transport, éducation, industrie

Chapitre 2 : Avantages et risques de l'IA

- Avantages de l'IA
- Risques de l'IA

Chapitre 3 : IA pour la recherche documentaire

- Présentation d'outils comme Elicit, Scopus AI, Web of Science Research Assistant.
- Rédaction de requêtes intelligentes
- Résumé automatique d'articles scientifiques
- Évaluation critique de sources proposées par IA

Chapitre 4 : IA pour la gestion du temps et des tâches

- Organisation personnelle (Notion AI, Google Calendar avec IA)
- Génération de to-do lists, création de plans de cours ou de révision

Chapitre 5 : IA pour la rédaction scientifique

- Aide à la structuration de devoirs, projets
- Reformulation, amélioration de style
- Attention au plagiat et à l'originalité
- Traduction et rédaction multilingue

Chapitre 6 : IA et créativité académique

- Génération d'idées de projets
- Brainstorming assisté par IA
- Création de présentations ou posters scientifiques

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

Bibliographie :

Cours en ligne :

1. *Elements of AI* (Université d'Helsinki) – Introduction accessible à l'IA .
2. *Objectif IA* (OpenClassrooms) – Cours pour tous sur les enjeux de l'IA .

Guides pratiques :

3. *Bibliothèque de l'Université Laval – Utilisation des outils d'IA en recherche documentaire .*
4. *Polytechnique Montréal – Outils d'IA générative en milieu universitaire*

Semestre:2

Unité d'enseignement : UED21

Matière : Electronique fondamentale

Crédits : 4

Coefficient : 2

Objectifs de l'enseignement :

Présenter les unités principales d'un ordinateur et expliquer leur fonctionnement ainsi que les principes de leur utilisation.

Connaissances préalables recommandées : Connaissances générales en informatique.

Contenu de la matière :

Chapitre 1 : Diodes à Semi-conducteur

- Semi-conducteur (SC) : généralités, SC intrinsèque, SC extrinsèque type P et type N, jonction PN non polarisée, jonction PN polarisée.
- La diode à jonction : polarisation et caractéristique de la diode, diode en continu, droite de charge et point de fonctionnement, diode en alternatif (redressement mono et double alternances, écrêtage, détecteur de crêtes).
- La diode Zener : définition, polarisation et caractéristique courant-tension, la diode Zener en continu (stabilisation de la tension) et en alternatif.

Chapitre 2 : Le transistor bipolaire

- Présentation, convention des courants (PNP et NPN), l'effet transistor, les courants du transistor, les différents montages d'un transistor (EC, CC, BC), caractéristiques de fonctionnement, polarisation du transistor (but et nécessité de polariser un transistor), droite d'attaque, droite de charge et point de fonctionnement, les circuits de polarisations, effet de la température et stabilité thermique. Le transistor en régime dynamique aux fréquences basses, droite de charge dynamique, le rendement en puissance et les classes d'amplificateurs, étude d'un amplificateur en émetteur commun, caractéristiques des amplificateurs EC, BC et CC, associations d'étages amplificateurs, le transistor en commutation.

Chapitre 3 : Transistors à effet de champ

Introduction, le transistor à effet de champ (TEC) à jonction (JFET): principe d'un transistor JFET, fonctionnement, caractéristiques, modèles électriques en amplification, montages amplificateurs (SC, DC, GC). Le transistor à effet de champ à grille isolée (MOSFET), MOSFET à appauvrissement, MOSFET à enrichissement, MOSFET en commutation, applications des MOSFET.

Chapitre 4 : Technologies des portes logiques

- Classification des circuits intégrés, caractéristiques des portes logiques, portes logiques DL, DTL, TTL et ECL, portes logiques NMOS et CMOS, la logique BiCMOS, comparaison entre la technologie bipolaire et CMOS, interfacement des portes logiques

Recommandations :

- Il est recommandé d'utiliser le vidéo projecteur pour le cours et de diffuser un support de cours ou photocopié.
- Utilisation d'un logiciel de simulation électronique (Workbench)

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

Références :

1. COEURDACIER S, « Amplification basses fréquences – commutation », Dunod, 1990.
2. LADJOUZE H, « Cours d'électronique », OPU, 25.
3. TAYEB CHERIF R. « Electronique de base », Berti éditions, 1990.
4. HARAUBIA B., « Electronique générale », OPU, 26.
5. BORNAND M., « Exercices et problèmes avec solutions », ELECTRONIQUE Tome 1, Vuibert, 1990.
6. COEURDACIER S, « Les composants discrets linéaires », ELECTRONIQUE 1, Dunod, 1990
7. MALVINO, « Principes d'électronique », Mc Graw-Hill, 24.
8. BORNAND M., « Problèmes d'électronique », ELECTRONIQUE Tome 2, Vuibert, 1990.

Semestre:3

Unité d'enseignement : UEF31

Matière : Architecture des ordinateurs

Crédits : 5

Coefficient : 3

Objectifs de l'enseignement : Ce module vise à fournir aux étudiants une compréhension approfondie de l'architecture des ordinateurs, en combinant des aspects théoriques et pratiques. Les objectifs pédagogiques incluent la compréhension des fondamentaux de l'architecture des ordinateurs, la maîtrise de l'architecture interne des processeurs, l'analyse et étude de cas des processeurs spécifiques et l'exploration des architectures de processeurs modernes.

Connaissances préalables recommandées : Structure machine.

Contenu de la matière :

Chapitre 1 : Organisation générale de l'unité centrale d'un ordinateur

1. Généralités sur l'Ordinateur
 - Historique des ordinateurs
 - Introduction aux concepts fondamentaux d'un ordinateur.
 - Composants essentiels : processeur, mémoire, dispositifs d'entrée/sortie.
2. Architecture de Base
 - Modèle de Harvard et modèle de Von Neumann : explication des deux modèles d'architecture et leurs utilisations respectives.
 - Processeur : rôle et composants principaux.
 - Mémoire : types de mémoire, rôles, et principes de base.
 - Bus : types de bus (données, adresse, contrôle) et leur importance pour la communication.

Chapitre 2 : Architecture interne des processeurs

1. Introduction
 - Présentation des éléments internes d'un processeur et de leur fonction.
2. Les Registres
 - Types de registres (données, adresse, contrôle).
 - Rôle des registres dans le traitement rapide des données.
3. Unité Arithmétique et Logique (UAL)
 - Rôle de l'UAL dans l'exécution des opérations arithmétiques et logiques.
 - Présentation des composants de l'UAL (additionneurs, comparateurs, etc.).
4. Unité de Commande (U.C)
 - Fonctionnement de l'unité de commande pour orchestrer l'exécution des instructions.
5. Jeu d'instruction
 - Ensemble des instructions qu'un processeur peut exécuter.
 - Catégories d'instructions : transfert de données, arithmétiques, logiques, contrôle de flux, etc.
6. Mode d'adressage
 - Techniques d'accès à la mémoire et aux données.
 - Différents types de modes d'adressage : immédiat, direct, indirect, indexé, etc.
7. Étapes d'exécution d'une instruction
 - Cycle d'instruction : phases de recherche, décodage, exécution, écriture.

Chapitre 3 : Étude de cas : processeur Intel 8086

1. Présentation de la famille 80x86 : historique, caractéristiques générales.
2. Architecture du Intel 8086
 - Registres spécifiques au 8086.
 - Adressage segmenté et modes d'adressage propres au 8086.
 - Jeu d'instructions : exemples et particularités du 8086.
3. Étapes de fonctionnement d'une instruction dans le 8086
 - Exemple d'une instruction et cycle d'exécution complet.
4. Jeu d'instructions détaillé
 - Instructions de transfert
 - Instructions arithmétiques
 - Instructions logiques
 - Contrôle de flux
 - Boucles et répétitions
 - Génération du code machine
5. Gestion des entrées/sorties
6. Gestion de la pile

7. Fonctions et sous-programmes
8. Interruptions

Chapitre 4 : Étude de cas : Processeurs 32 et 64 bits

1. Introduction aux architectures 32 bits et 64 bits
 - Différences principales (capacité mémoire, performances, compatibilité).
 - Évolution des architectures (IA-32 à AMD64).
2. Registres dans les processeurs 32 et 64 bits
 - Registres généraux, pointeurs, et registres de segment.
 - Extensions des registres en 64 bits.
3. Modes d'adressage avancés
 - Passage de l'adressage segmenté à l'adressage plat.
 - Adressage base + offset, indexé, relatif (spécificités 64 bits).
4. Jeu d'instructions
 - Particularités et extensions modernes : SSE, AVX.
 - Exemples comparés entre instructions 32 et 64 bits.
5. Cycle d'exécution d'une instruction
 - Étapes du pipeline pour une instruction donnée.
 - Exemple : MOV, ADD ou JMP en 64 bits.

Chapitre 5 : Architectures des processeurs récents

- Introduction aux processeurs modernes : tendances et évolutions technologiques.
- Multi-cœurs et parallélisme
 - Concept de multi-cœur et traitement parallèle.
 - Avantages et défis des architectures multi-cœurs.
- Mémoire cache et hiérarchie mémoire
 - Gestion de la mémoire cache dans les architectures récentes.
 - Hiérarchies mémoire pour optimiser les temps d'accès.

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

Références :

1. Structured Computer Organization, Auteur: Andrew S. Tanenbaum, Édition : Pearson, 2012
2. Computer Organization and Design: The Hardware/Software Interface; Auteurs: David A. Patterson, John L. Hennessy Edition : Morgan Kaufmann.
3. Architecture et technologie des ordinateurs Cours et exercices corrigés - 6e édition, Paolo Zanella, Yves Ligier, Emmanuel Lazard, 2018
4. The Intel Microprocessors: 8086/8088, 80186/80188, 80286, 80386, 80486, Pentium, Pentium Pro Processor, Pentium II, Pentium III, Pentium 4, and Core2 with 64-bit Extensions, Auteur : Barry B. Brey Édition : Pearson
5. EMU8086 -MICROPROCESSOR EMULATOR <https://emu8086-microprocessor-emulator.fr.softonic.com/>
6. NASM (Netwide Assembler) <https://www.nasm.us/>
7. MASM (Microsoft Macro Assembler) <https://learn.microsoft.com/en-us/cpp/assembler/masm/microsoft-macro-assembler-reference?view=msvc-170>
8. GAS (GNU Assembler) <https://www.gnu.org/software/binutils/>
9. YASM <https://yasm.tortall.net/>

Semestre:3

Unité d'enseignement : UEF31

Matière : Algorithmique et structure de données 3

Crédits : 6

Coefficient : 5

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 :

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

1. Thomas H. Cormen, Algorithmes Notions de base *Collection : Sciences Sup, Dunod*, 2013.
2. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest *Algorithmique - 3ème édition - Cours avec 957 exercices et 158 problèmes Broché*, Dunod, 2010.
3. Rémy Malgouyres, Rita Zrour et Fabien Feschet. *Initiation à l'algorithmique et à la programmation en C : cours avec 129 exercices corrigés*. 2^{ème} Edition. Dunod, Paris, 2011. ISBN : 978-2-10-055703-5.
4. Damien Berthet et Vincent Labatut. *Algorithmique & programmation en langage C - vol.1 : Supports de cours*. Licence. Algorithmique et Programmation, Istanbul, Turquie. 2014, pp.232.
5. Damien Berthet et Vincent Labatut. *Algorithmique & programmation en langage C - vol.2 : Sujets de travaux pratiques*. Licence. Algorithmique et Programmation, Istanbul, Turquie. 2014, pp.258. <cel-01176120>
6. Damien Berthet et Vincent Labatut. *Algorithmique & programmation en langage C - vol.3 : Corrigés de travaux pratiques*. Licence. Algorithmique et Programmation, Istanbul, Turquie. 2014, pp.217. <cel-01176121>
7. Claude Delannoy. *Apprendre à programmer en Turbo C*. Chihab- EYROLLES, 1994.

Semestre:3

Unité d'enseignement : UEF32

Matière : Systèmes d'information

Crédits : 4

Coefficient : 2

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^[1]_{SEP} (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, 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
- Modélisation:
 - Statique (modèle conceptuel de données : MCD).
 - Dynamique (modèle conceptuel de traitement : MCT).

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

Références

1. Coord. P. Vidal, P. Planeix, Systèmes d'information organisationnels, 25.
2. Coord. M-L. Caron-Fasan & N. Lesca, Présents et futurs des systèmes d'information, 23, PUG. p.
3. Kalika M. & alii, Le e-management. Quelles transformations pour l'entreprise ? , 23, Editions Liaisons.
4. J.L.Lemoigne, La théorie du système général. PUF-
5. V. Bertalanfy, Théorie générale des systèmes. Dunod.^[1]_{SEP}
6. X. Castellani, Méthode générale d'analyse d'une application informatique. Masson, 1975.
7. Tardieu et al. , « la méthode merise : principes et outils », éd. d'organisation, 1983.-
8. Tardieu et al. , « la méthode merise : démarche et pratique » éd. d'organisation, 1985.-
9. Tabourier, « de l'autre côté de Merise », éd. d'organisation, 1986.-
10. J. P. Mathéron, « Comprendre Merise », 1990

Semestre: 3

Unité d'enseignement : UEF32

Matière : Programmation orientée objet 1

Crédits : 5

Coefficient : 3

Objectifs de l'enseignement :

- Initiation au paradigme orienté objet (principes et mécanismes).
- Manipulation des concepts de l'orienté objet.
- Conception de solutions orientées objet et mise en œuvre en java.
- Maîtrise des éléments de base du langage java.

Objectifs du TP :

- Utiliser une partie des séances de TP pour les TD
- Pratique et mise en œuvre des concepts vus en cours sur machine.
- Un projet de TP (ou plus) englobant des aspects conceptuels et pratiques, est nécessaire pour permettre aux étudiants de mieux comprendre les concepts étudiés.

Connaissances préalables recommandées : Connaissances de base en Algorithmique.

Contenu de la matière :

Chapitre 1 : Introduction à la POO

1. De l'impératif à l'objet : l'évolution des paradigmes de programmation
2. Fondements et philosophie de la POO : modélisation du monde réel, encapsulation, abstraction, réutilisation, modularisation et facilitation de la maintenance du code
 - Regroupement des traitements et données en POO.
 - Encapsulation.
 - Abstraction.

Chapitre 2 : Concepts de base de la POO

1. Notion de classe
 - Structure d'une classe : attributs et méthodes
 - Constructeurs des classes et initialisation des attributs.
 - Encapsulation, portée des attributs et droits d'accès.
 - Champs d'instance et champs de classe
2. Notion d'objet
 - Définition et rôle d'un objet
 - Création et manipulation d'objets
 - Instanciation via le mot-clé new
 - Appel de méthodes et accès aux attributs
 - Cycle de vie d'un objet
 - Références et objets
 - Assignment d'objets à des variables
 - Comparaison d'objets (== vs .equals())
 - Relations entre objets
 - Association
 - Agrégation
 - Composition
 - Interaction entre objets
 - Appel de méthodes d'un autre objet
 - Collaboration objet-objet
 - Passage d'objets en paramètres

Chapitre 3 : Héritage et polymorphisme

1. Héritage.
 - Définition et principe de l'héritage
 - Héritage simple (cas de Java)
 - Héritage multiple : gestion des conflits et résolution du problème du diamant (cas de C++ et Python)
2. Polymorphisme
 - Définition et principe de polymorphisme
 - Polymorphisme à l'exécution (late binding)
 - Polymorphisme à la compilation (early binding)
3. Classe Object
 - Rôle de la classe Object en Java

- Méthodes de la classe Object et leur héritage par défaut
- Impact sur les concepts de polymorphisme et d'héritage

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

Références :

1. B. Meyer, « Conception et programmation orientée objet », Edition Eyrolles, 28.
2. 2- F. Barbier, « Conception orientée objet en Java et C++ : une approche comparative », Editions Pearson Education, Septembre 29.
3. C. Delannoy « Programmer en JAVA », Editions Eyrolles, 9eme Edition 2016.
4. Hugues Bersini, « La programmation orientée objet, cours et exercices en UML2 avec Java 5, C# 2, C++, Python, PHP 5 et LINQ », Editions Eyrolles, 29.

Semestre:3

Unité d'enseignement Méthodologique : UEM31

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 (60%), contrôle continu (40%).

Références :

1. M. Schatzman Analyse numérique : une approche mathématique, Dunod 24.
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 20 ;
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.

Semestre:3

Unité d'enseignement : UEM31

Matière : probabilités et statistique 1

Crédits : 4

Coefficient : 2

Objectifs de l'enseignement :

Introduire les notions fondamentales en probabilités et en séries statistiques à une variable et à deux variables.

Connaissances préalables recommandées : Mathématiques de base

Contenu de la matière :

Chapitre 1 : Notions de base et vocabulaire statistique

- Concepts de base de la statistique (Population et individu, Variable (ou caractère))
- Les tableaux statistiques : Cas de variables qualitatives (Représentation circulaire par des secteurs, Représentation en tuyaux d'orgue, Diagramme en bandes), cas de variables quantitatives (Le diagramme en bâtons, Histogramme, Polygone).

Chapitre 2 : Représentation numérique des données

- Les caractéristiques de tendance centrale ou de position (La Médiane, Les quartiles, Intervalle interquartile, Le mode, La moyenne arithmétique, La moyenne arithmétique pondérée, La moyenne géométrique, La moyenne harmonique, La moyenne quadratique).
- Les caractéristiques de dispersion (L'étendu, L'écart type, L'écart absolue moyen, Le coefficient de variation).

Chapitre3 : Calcul des probabilités

- a. Analyse combinatoire : (Principe fondamental de l'analyse combinatoire, Arrangements, Permutations, Combinaisons).
- b. Espace probabilisable : (Expérience aléatoire, Evénements élémentaires et composés, Réalisation d'un événement, Evénement incompatible, Système complet d'événement, Algèbre des événements, Espace probabilisable, Concept de probabilité).
- c. Espace probabilisé : (Définitions, conséquence de la définition, probabilité conditionnelle, événements indépendants, expériences indépendantes)
- d. Construction d'une probabilité
- e. Probabilités conditionnelles, indépendance et probabilités composées (Probabilités conditionnelles, Indépendance, Indépendance mutuelle, Probabilités composés, Formule de Bayes).

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

Références

1. G. Calot, Cours de statistique descriptive, Dunod, Paris, 1973.
2. P. Bailly, Exercices corrigés de statistique descriptive, OPU Alger, 1993.
3. H. Hamdani, Statistique descriptive avec initiation aux méthodes d'analyse de l'information économique: exercices et corrigés, OPU Alger, 26.
4. K. Redjda, Probabilités, OPU Alger, 24

Semestre:3

Unité d'enseignement : UET31

Matière : Gestion de projets

Crédits : 2

Coefficient : 1

Mode d'enseignement : Présentiel

Objectifs d'enseignement :

Ce cours développe les compétences et l'expertise en conception, planification et contrôle de projets. Il examine le cycle de vie de la gestion de projet, la définition des paramètres, les défis de gestion, les outils et techniques de gestion de projet, et met l'accent sur le rôle du chef de projet.

Il guidera les étudiants à travers les outils fondamentaux de la gestion de projet et les compétences nécessaires pour concevoir, lancer, diriger et mettre en œuvre des projets réussis. Les chefs de projet performants possèdent les compétences nécessaires pour gérer leurs équipes, leurs calendriers et leurs ressources afin d'obtenir les résultats escomptés.

Connaissances préalables recommandées :

Contenu de la matière :

1) Introduction à la gestion de projet

- Définition du projet
- Cycle de vie du projet
- Le chef de projet

2) Organisation du projet

- Organisation fonctionnelle,
- Equipes dédiées,
- Organisation matricielle

3) Définition du projet

- Définition des frontières du projet
- Établissement des priorités du projet
- Création de l'organigramme des tâches (Work Breakdown Structure)
- Intégration de l'organigramme des tâches à l'organisation
- Codage de l'organigramme des tâches pour le système d'information
- Matrices de responsabilités

4) Estimation des délais et des coûts du projet

- Facteurs influençant la qualité des estimations
- Directives d'estimation des délais, des coûts et des ressources
- Estimation descendante et ascendante
- Méthodes d'estimation des délais et des coûts du projet

5) Élaboration d'un plan de projet

- Construction d'un réseau de projet
- Principes fondamentaux de l'activité sur nœud
- Processus de construction du réseau

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

Références

1. Project management : the managerial process /
Erik W. Larson, Clifford F. Gray, Seventh edition, McGraw-Hill Education, 2018
2. CNRS, DSI (<http://www.dsi.cnrs.fr/conduite-projet/Default.htm>)
3. Association Francophone de la gestion de projet (<http://www.afitep.fr/Default.htm>)
4. Project Management Institute (PMI) (<http://www.pmi.org/>)
5. Software Engineering Institute (SEI) (<http://www.sei.cmu.edu/>)
6. IEEE Software Engineering Group (<http://standards.ieee.org/software/>)
7. Guide to the Software Engineering Body of Knowledge (<http://www.swebok.org/>)
8. Choosing a project management tool
 - <http://www.4pm.com/articles/selpmsw.html>
 - <http://www.infogoal.com/pmc/pmcswr.htm>
 - Project management tools <http://www.startwright.com/project1.htm>

Semestre:4

Unité d'enseignement : UEF41

Matière : Système d'exploitation 1

Crédits : 5

Coefficient : 4

Objectifs de l'enseignement : Introduire les notions de base des systèmes d'exploitation, leurs rôles dans la gestion des ressources de la machine : processeur et mémoire centrale puis présenter les mécanismes et les techniques utilisées à ces fins.

Connaissances préalables recommandées : algorithmique et structures de données, structure machine.

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 Siamoises (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.

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, third 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 , Dunod, collection : Sciences sup, 24.
4. Crocus, Systèmes d'exploitation des ordinateurs, Dunod, 1993.
5. Sacha Krakowiak, Principes des systèmes d'exploitation des ordinateurs, Dunod, 1993

Semestre:4

Unité d'enseignement : UEF41

Matière : Théorie des langages

Crédits : 5

Coefficient : 3

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. 26, Dunod.
2. P. Séébold. Théorie des automates. 29, 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: 4

Unité d'enseignement : UEF42

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)
 - d. Schéma de base de données
5. Modèle relationnel logique (SQL)
 - a. Table, colonne, et ligne
 - b. Description de SQL (StructuredQueryLanguage)
 - 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 : Union, Intersection, Différence, Produit cartésien, Jointure (Thêta, naturelle, équijointure, externe), Division
4. 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 23
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:4

Unité d'enseignement : UEF42

Matière : Programmation orientée objet 2

Crédits : 5

Coefficient : 3

Objectifs de l'enseignement :

- Approfondissement et consolidation des concepts fondamentaux de la POO étudiés en POO I (S3).
- Maîtrise des principes et techniques avancées de la POO.
- Développement de compétences pour concevoir et implémenter des applications complexes, modulaires, réutilisables et évolutifs en Java.
- Maîtrise du développement d'applications entièrement orientées objet, incluant les interfaces graphiques, l'intégration des bases de données et la gestion des objets distribués.

Objectifs du TP :

- Utiliser une partie des séances de TP pour les TD
- Pratique et mise en œuvre des concepts vus en cours sur machine.
- Un projet de TP (ou plus) englobant des aspects conceptuels et pratiques, est nécessaire pour permettre aux étudiants de mieux comprendre les concepts étudiés.

Connaissances préalables recommandées : Connaissances de base en programmation Orientée Objet.

Contenu de la matière :

Chapitre 0 : Rappels sur le paradigme orienté objet (POO1).

Chapitre 1 : Structures avancées et modularité en POO

1. Classes abstraites
2. Interfaces
3. Classes internes
4. Classes anonymes
5. Notion de package (paquetage)
6. Exemples de packages utiles en java : : java.lang, java.util
7. Organisation des classes en packages

Chapitre 2 : Gestion des exceptions

1. Définition d'exception
2. Les différents types d'erreurs
3. Hiérarchie des classes d'exception
4. Gestion des exceptions (bloc try ...catch)
5. Arrêt sélectif des erreurs
 1. Plusieurs exceptions dans un bloc try...catch

Chapitre 3 : Flux d'entrées/sorties

1. Gestion des fichiers
 - a. La classe File
 - b. Méthodes de la classe File
 - c. Filtre sur les noms de fichiers
2. Définition générale d'un flux
3. Classification des flux : entrée vs sortie, caractère vs binaire
4. Flux d'écran
 - Utilisation de la sortie standard
 - Formatage des sorties
5. Flux de clavier
 - Lecture simple avec System.in
 - Lecture typée avec la classe Scanner
6. Flux de fichiers
 - Ouverture et fermeture de fichiers
 - Écriture dans un fichier texte
 - Lecture caractère par caractère et ligne par ligne
7. Redirection de flux
8. Hiérarchie des classes Java pour les flux d'E/S binaires

Chapitre 4 : Programmation événementielle et Interfaces graphiques

1. Généralités sur les interfaces graphiques.
2. Composants des interfaces graphiques.
3. Conteneurs et écouteurs de composants
4. Packages AWT et Swing.

- Classes de base.
 - Création et affichage d'une fenêtre.
 - Disposition des composants dans une fenêtre.
 - Gestion des événements.
5. Introduction à JavaFX et sa différence avec AWT/Swing
- Hiérarchie des objets graphiques en JavaFX
 - Les objets graphiques de base (Stage, Scene, Gestionnaires de disposition)
 - Types d'évènements
 - Écouteurs (simples, écouteurs partagés, et auto-écouteurs)
 - Gestionnaires des événements
 - Propagation et filtrage des événements.

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

Références

1. B. Meyer, « Conception et programmation orientée objet », Edition Eyrolles, 28.
2. F. Barbier, « Conception orientée objet en Java et C++ : une approche comparative », Editions Pearson Education, Septembre 29.
3. C. Delannoy « Programmer en JAVA », Editions Eyrolles, 9eme Edition 2016.
4. Hugues Bersini, « La programmation orientée objet, cours et exercices en UML2 avec Java 5, C# 2, C++, Python, PHP 5 et LINQ », Editions Eyrolles, 29.
5. Kasprian Raffi, Java For Artists: The Art, Philosophy, and Science of Object-Oriented Programming, Pulp Free Press, 26.
6. John R. Hubbard, Structures de données en Java, Ediscience, 23
7. Michael B. White, Mastering Java: An Effective Project Based Approach including Web Development, Data Structures, GUI Programming and Object-Oriented Programming (Beginner to Advanced), Newstone, 2018.
8. Boyarsky Jeanne, Selikoff Scott, OCP: Oracle Certified Professional Java SE 8 Programmer II Study Guide, Sybex, 2015.

Semestre:4

Unité d'enseignement : UEM41

Matière : Probabilités et statistique 2

Crédits : 4

Coefficient :2

Objectifs de l'enseignement : ce cours constitue une introduction à l'étude des modèles aléatoires simples. L'objectif est de fournir les outils indispensables dans le domaine des probabilités, et également d'aborder les aspects statistiques. À la fin de ce module, l'étudiant devrait être en mesure de calculer les différentes mesures de dispersions dans les statistiques et d'effectuer des probabilités basées sur les lois de la probabilité et de faire des tests sur des données en utilisant les théories de la probabilité.

Connaissances préalables recommandées :

Contenu de la matière :

1. Espaces probabilisés
2. Variables aléatoires discrètes
3. Variables aléatoires continues
4. Fonctions caractéristiques
5. Théorèmes limites
6. Vecteurs gaussiens
7. Simulation
8. Estimateurs
9. Intervalle et régions de confiance
10. Tests

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

Références

1. Lecoutre B., Tassi Ph. (1987) Statistique non paramétrique et robustesse Paris : Economica.
2. Tassi Ph. (1989) Méthodes statistiques Paris: Economica –
3. Tassi Ph., Legait S. (1990) Théorie des probabilités en vue des applications statistiques Paris : Ed. Technip
4. Saporta, G., Probabilités, Analyse des données et Statistique, Technip, 2ème édition, 26
5. Jean-Pierre Lecoutre, Statistique et probabilités, Editions Dunod, 2012.
6. Yadolah Dodge, Valentin Rousson, Analyse de régression appliquée, Editions Dunod, 24.

Semestre: 4

Unité d'enseignement : UEM41

Matière : Programmation linéaire

Crédits : 4

Coefficient : 2

Objectifs de l'enseignement : Ce module a pour objectifs de sensibiliser l'étudiant à l'importance pratique des problèmes d'optimisation linéaires, de maîtriser l'ensemble théorique sous-jacent, et de pouvoir utiliser ces techniques dans des problèmes pratiques.

Connaissances préalables recommandées : Mathématiques et informatique générales

Contenu de la matière :

Chapitre1 : Introduction générale

- 1.1 Historique de la programmation linéaire
- 1.2 Exemples de modélisation de problèmes pratiques sous forme de programme linéaire.

Chapitre2 : Géométrie de la programmation linéaire

- 2.1 Espaces vectoriels, rang de matrice, systèmes d'équations linéaires
- 2.2 Ensemble convexe, hyperplan, polyèdre, simplexe, point extrême

Chapitre3 : Méthode primale de résolution d'un programme linéaire

- 3.1 Position du problème
- 3.2 Caractérisation des points extrêmes
- 3.3 Optimalité en un point extrême
- 3.4 Critères d'optimalité : formule d'accroissement de la fonction objectif, critère d'optimalité, 3.5 condition suffisante d'existence de solution non bornée
- 3.6 Algorithme du simplexe : amélioration de la fonction objectif en passant d'un point extrême à un autre, algorithme du simplexe sous forme matricielle, finitude de l'algorithme du simplexe, algorithme et tableau du simplexe
- 3.7 Initiation de l'algorithme du simplexe : cas du programme linéaire sous forme normale, M-méthode, méthode de deux phases,

Chapitre4 : Méthodes duales en programmation linéaire

- 4.1 Définitions
- 4.2 Formule d'accroissement de la fonction duale et critère d'optimalité
- 4.3 Condition suffisante de solutions réalisables dans le problème primale
- 4.4 Algorithme dual du simplexe
- Initialisation de l'algorithme duale du simplexe

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

Références

- 1. M. Sakarovich, Graphes et programmation linéaire, Ed. Hermann, 1984.
- 2. H. Mauran, Programmation linéaire appliquée, Ed. Technip, 1967.
- 3. A. Kauffman, Méthodes et modèles de R.O., Ed. Dunod, 1976.
- 4. V. Chvatal, Linear programming. W.H. Freeman and Company, 1983.

Semestre:4

Unité d'enseignement : UED31

Matière : Entreprenariat

Crédits : 2

Coefficient : 1

Objectifs de l'enseignement :

Ce cours a pour objectif d'aider l'étudiant à structurer, démarrer et développer son projet entrepreneurial. Le but du cours est aussi de développer chez l'étudiant le mécanisme « Apprendre à Entreprendre ». L'objectif consiste aussi à développer chez l'étudiant la créativité entrepreneuriale via une mise en valeur d'une idée par des projets de type « business model ». Le module repose sur quatre points qui sont :

- Motiver l'étudiant à la création d'une startup,
- Maîtriser des outils de formalisation et de mise en œuvre d'un projet de startup
- Transformer les bonnes idées de création d'un modèle d'affaire (business model),
- Adapter l'étudiant à l'écosystème et à la culture des startups.

Contenu de la matière :

Chapitre 1 : Introduction à l'Entreprenariat

- Définitions et importance de l'entrepreneuriat
- Les défis et opportunités du monde entrepreneurial

Chapitre 2 : Le Profil Entrepreneurial

- Les compétences essentielles d'un entrepreneur
- Se lancer seul ou en équipe : avantages et inconvénients
- Exemples d'entrepreneurs inspirants et leurs parcours

Chapitre 3 : Le processus entrepreneurial

Comprendre l'Écosystème Startup & Le Cycle de Vie d'une Startup

- o Les acteurs clés : incubateurs, accélérateurs, business angels, VC, ...
- o L'importance du réseau et du mentorat
- o Les différentes phases de développement d'une startup
- Identifier les Opportunités et Trouver l'Idée Gagnante
 - o Techniques de créativité et génération d'idées
 - o Études de marché et validation du besoin
 - o Outils et méthodes pour tester son idée
 - o Protéger l'idée (propriété intellectuelle, Brevet, NDA, ...)
- Le Marketing et le Développement de la Clientèle
 - o Comment identifier et attirer ses premiers clients ?
 - o Stratégies d'acquisition et fidélisation
 - o Techniques de Growth Hacking pour accélérer la croissance
- Business Model Canvas (BMC) et Business Plan
 - o Structure d'un Business Model Canvas
 - o Structure d'un Business Plan

Chapitre 4 : Aspects juridiques et financiers d'une startup

Le Financement de la Startup

- o Les différentes options de financement : Business Angels, Venture Capital, Crowdfunding, ...
- Gestion Juridique et Administrative
 - o Choix du statut juridique (auto-entrepreneur, SARL, ...)
 - o Fiscalité et obligations légales
 - o Contrats et accords

Chapitre 5 : Projet de groupe (mini-startup à présenter en fin de module)

Mode d'évaluation : Examen (100%)

Références :

1. Robert Papin, La création d'entreprise, Création, reprise, développement, 16e édition - Collection : Hors collection, Dunod, 2015.
2. Eric Ries, Lean Startup : Adoptez l'innovation continue, Éditeur : PEARSON, 2015.
3. Vincent Ydé, Créer son entreprise : du projet à la réalité, Éditeur : VUIBERT, 29.

Semestre: 5

Unité d'enseignement : UEF51

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 : Structure machine, composants et systèmes.

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." (27).
2. Tanenbaum, Andrew S. "Computer networks, 4-th edition." ed: Prentice Hall (23).

Semestre:5

Unité d'enseignement : UEF51

Matière : Compilation

Crédits :5

Coefficient : 3

Objectifs de l'enseignement : Comprendre comment les programmes sont compilés puis exécutés

Connaissances préalables recommandées : Théorie des langages

Contenu de la matière :

Chapitre 1: Introduction (objectifs ...)

Chapitre 2 : Compilation

- i. Définition d'un compilateur
- ii. Structure d'un compilateur

Chapitre 3 : Analyse lexicale

Chapitre 4 : Analyse Syntaxique

- i. Dérivation la plus à gauche et arbre de dérivation
- ii. Grammaire ambiguë
- iii. Grammaire et langages de programmation
- iv. Analyseurs syntaxiques et leurs types
- v. Outils en pratique

Chapitre 5 : Analyse descendante

- i. Analyse LL(1) (principe)
- ii. Table d'analyse
- iii. Grammaire LL(1)

Chapitre 6 : Analyse ascendante

- i. Analyse LR (principe)
- ii. Analyse LR(0)
- iii. Analyse SLR(1)
- iv. Analyse LR(1)AnalyseLALR(1)

Chapitre 7 : Traduction dirigée par la syntaxe

Chapitre 8 : Contrôle de type

Chapitre 9:Environnement d'exécution

Chapitre 10 : Génération de code

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

Références

1. Alfred Aho, Ravi Sethi et Jeffrey Ullman « Compilers, Principles techniques and tools » Addison-Wesley 1986

Semestre:5

Unité d'enseignement : UEF52

Matière : Génie logiciel

Crédits : 5

Coefficient : 3

Objectifs de l'enseignement : Apprendre à appliquer une méthodologie d'analyse et de conception pour le développement des logiciels. En particulier, apprendre la modélisation objet avec le langage universel UML.

Connaissances préalables recommandées : Algorithmique, Système d'Information, Programmation Orientée Objet.

Contenu de la matière :

Chapitre 1. Introduction

1. Définitions et objectifs
2. Principes du Génie Logiciel
3. Qualités attendues d'un logiciel
4. Cycle de vie d'un logiciel
5. Modèles de cycle de vie d'un logiciel

Chapitre 2. Modélisation avec UML

1. Introduction
Modélisation, Modèle, Modélisation Orientée Objet, UML en application.
2. Eléments et mécanismes généraux
3. Les diagrammes UML
4. Paquetages

Chapitre 3. Diagramme UML de cas d'utilisation : vue fonctionnelle

Intérêt et définition, Notation

Chapitre 4. Diagrammes UML de classes et d'objets : vue statique

1. Diagramme de classes
2. Diagramme d'objets

Chapitre 5. Diagrammes UML : vue dynamique

1. Diagramme d'interaction (Séquence et collaboration)
2. Diagramme d'activités
3. Diagramme d'états/transitions

Chapitre 6. Autres notions et diagrammes UML

1. Composants, déploiement, structures composite.
2. Mécanismes d'extension : langage OCL + les profils.

Chapitre 7. Introduction aux méthodes de développement : (RUP, XP) et Rétro-ingénierie

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

Références :

1. Bern Bruegge and Allen H. Dutoit, *Object-Oriented Software Engineering – using UML, Patterns and Java*. Third Edition, Pearson, 2010.
2. G. Booch, J. Rumbaugh, I. Jacobson, "The Unified Modeling Language (UML) Reference Guide", Addison-Wesley, 1999.
3. G. Booch, J. Rumbaugh, I. Jacobson, "The Unified Modeling Language (UML) User Guide", Addison-Wesley, 1999.
4. G. Booch et al., "Object-Oriented Analysis and Design, with applications", Addison- Wesley, 27.
5. Laurent Audibert. Cours UML 2.0, disponible sur <http://www.developpez.com>.
6. M. Blaha et J. Rumbaugh. *Modélisation et conception orientées objet avec UML 2*. 2ème édition. Pearson Education, 25.
7. Pierre-Alain Muller. *Modélisation objet avec UML*. Éditions Eyrolles, 23.
8. Shari Lawrence Pfleeger and Joanne M. Atlee. *Software Engineering*. Fourth Edition, Pearson, 2010.

Semestre:5

Unité d'enseignement : UEF52

Matière : Développement d'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'algorithmique 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%)

Références

1. Cours de développement Web. Disponible sur le site : <https://openclassrooms.com/courses>.
2. Jean ENGELS. "*PHP 5 – Cours et exercices*". Editions Eyrolles, 25
3. Mathieu Lacroix, "*Introduction Web: Cours*". Université paris 13, 2013.
4. Société Digimind. "*Le Web 2.0 pour la veille et la recherche d'information, Exploitez les ressources du web social*", Digimind, juin 27.

Semestre:5

Unité d'enseignement : UEM51

Matière : Outils de programmation scientifique

Crédits : 2

Coefficient : 1

Objectifs de l'enseignement :

Maîtrise des outils de programmation scientifique

Connaissances préalables recommandées : Notions de base d'algorithmique et de programmation.

Contenu de la matière :

Chapitre 1 : Langages de programmation (Python, R...)

Chapitre 2 : Maîtrise de Logiciels (Scilab, Matlab..)

Chapitre 3 : Exemples d'applications et techniques de résolution

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

Références

1. Data Analysis Software: Gnu Octave, Mathematica, MATLAB, Maple, Scilab, Social Network Analysis Software, LabVIEW, Eicaslab. 2010. Editeur Books LLC., 2010.
2. J.T. Lapresté., Outils mathématiques pour l'étudiant, l'ingénieur et le chercheur avec Matlab, 28; Editeur ellipses.
3. Grenier Jean-Pierre, Débuter en Algorithmique avec MATLAB et SCILAB, Editeur ellipses, 27

Semestre:5

Unité d'enseignement : UEM51

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

- 1.1. 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 extréum
 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, (3 pages).
2. Nguyen Huy Xuong, Mathématiques discrètes et informatique, Masson, 1997
3. Aimé Sacher, La théorie des graphes, Que-Sais-Je ?, 1974 ; réédition prévue en 24 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, 20.
7. Bojan Mohar, Carsten Thomassen, Graphs on surfaces, John Hopkins University Press, Baltimore, 21.

Semestre: 5

Unité d'enseignement : UET51

Matière : Rédaction scientifique

Crédits : 2

Coefficient : 1

Objectifs de l'enseignement : apprendre les techniques de rédaction scientifique pour permettre à l'étudiant comment à publier les résultats de leurs travaux de recherche.

Connaissances préalables recommandées : connaissance sur des travaux scientifiques

Contenu de la matière :

Introduction

Avant la rédaction

- a. Organisation des idées et plan de la rédaction
- b. Préparer la manière de rédiger
- c. Choisir ses outils de mise en page
- d. Systèmes de contrôle des versions

Références bibliographiques

- a. Obtenir des références
- b. Construire la bibliographie
- c. Eviter le plagiat

Rédaction

- a. Structure
- b. Contenu
- c. Style
- d. Présentation

Mode d'évaluation : contrôle continu (100 %)

Références :

1. BRUYERE, V. Comment bien rédiger. Exposé ; a l'intention des étudiants de 1ere licence en informatique, Université de Mons-Hainaut, 26.
2. VALDURIEZ, P. Some Hints to Improve Writing of Technical Papers. Ingénierie des Systèmes d'Informations 2, 3 (1994), 371–375.

Semestre: 6

Unité d'enseignement : UEF61

Matière : Système d'exploitation 2

Crédits : 5

Coefficient : 4

Objectifs de l'enseignement : Une étude approfondie du système Unix est recommandée pendant les séances de TD et de TP. La programmation des threads et des mécanismes de l'exclusion mutuelle se fera en C sous Unix. Les modèles producteur/consommateur, lecteur/rédacteurs et des philosophes avec plusieurs variantes seront étudiés de façon théorique (développement d'algorithmes en pseudo-langage) en TD puis implémentés en C sous Unix durant les séances de TP.

Connaissances préalables recommandées : système d'exploitation 1.

Contenu de la matière :

Chapitre 1 :

- Rappels sur la notion de SE.
- Notions de programme, processus, thread et ressource partagée.

Chapitre 2 : Synchronisation de processus.

- Problème de l'accès concurrent à des ressources et sections critiques (Problème de l'exclusion mutuelle)
- Outils de synchronisation :
 - Événements, Verrous
 - Sémaphores
 - Moniteurs
 - Régions critiques.
 - Expressions de chemins

Chapitre 3 : La communication interprocessus

- Partage de variables (modèles : producteur/ consommateur, lecteurs/ rédacteurs)
- Boîtes aux lettres
- Echange de messages (modèle du client/ serveur)

Chapitre 4 : L'inter blocage

- Modèles
- Prévention
- Evitement
- Détection/ Guérison

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

Références

1. Tanenbaum, Modern operating systems, third 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 , 24.
4. Crocus, Systèmes d'exploitation des ordinateurs, 1993.
5. Sacha Krakowiak, Principes des systèmes d'exploitation des ordinateurs, Dunod, 1993

Semestre: 6

Unité d'enseignement : UEF61

Matière : Sécurité Informatique

Crédits : 5

Coefficient : 3

Objectifs de l'enseignement : cette matière permet aux étudiants d'acquérir des compétences pour assurer la sécurité et le bon fonctionnement des systèmes informatiques.

Connaissances préalables recommandées : fondement algorithmique, technique de programmation

Contenu de la matière :

Chapitre I : Introduction à la sécurité

I.1 - Définitions : Sécurité, Sureté de fonctionnement, ...

I.2 - Principaux concepts de sécurité informatique

- vulnérabilité , - menace , - contre-mesure , risque, ...

I.3- Objectifs de la sécurité informatique

- La confidentialité, L'intégrité, La disponibilité, La non-répudiation, L'authentification, ...

I.4- Les menaces informatiques

- Qu'est-ce qu'une attaque ?

- Définitions : Virus - Ver - Cheval de Troie - Logiciel espion

- Origine des attaques

- Qui peut être visé ?

- Les étapes d'une attaque

- Les différentes taxonomies des attaques

- Les différents types d'attaques :

- Les attaques réseaux - Les attaques de système -Les attaques de mots de passe

- Attaque de site web - Attaque d'applications.

- Moyens de lancer une attaque

I.5- Méthodes de défense

- Anti-virus, - Pare-feux, - Réseaux privés, - Détection d'intrusion, etc...

Chapitre II : Initiation à la cryptographie

II.1- Vocabulaire et définitions

- Cryptologie,-Cryptographie ,Cryptogramme , -Cryptanalyse, etc...

II.2- Histoire de la cryptographie

II.3- Cryptographie Classique

- Algorithme de Substitution : Chiffre de César, chiffre de VIGENERE.

- Algorithme de Transposition : la technique assyrienne.

II.4- Cryptographie Moderne

- Cryptographie symétrique : Principe, Algorithmes DES et AES

- Cryptographie Asymétrique : Principe, Algorithme RSA

II.5- Fonctions de Hachage

- Principe

- Les algorithmes MD5 et SHA-1.

II.6- La signature électronique

II.7- Les certificats numériques

II.8- Autorités de certification et PKI

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

Références :

1. Laurent Bloch , Christophe Wolfhugel , Ary Kokos ,Gérôme Billois , Arnaud Soullié , Alexandre Anzala-Yamajako , Thomas Debize, Sécurité informatique pour les DSI, RSSI et administrateurs, éditions Eyrolles , 5° édition, Collection Blanche, 2016.
2. Jean-François Pillou, Jean-Philippe Bay, Tout sur la sécurité informatique, DUNOD, 4° ÉDITION, 2016.
3. Gilles Dubertret, L'univers secret de la cryptographie, Vuibert, 2015.
4. Damien Vergnaud, Exercices et problèmes de cryptographie ,*Collection : Sciences Sup, Dunod*, 2015

Semestre: 6

Unité d'enseignement : UEF62

Matière : Recherche d'information

Crédits : 5

Coefficient : 3

Objectifs de l'enseignement : Cette matière a pour objectif de donner à l'étudiant les principes de base avec les approches pour la recherche d'information à partir d'une grande masse d'informations. L'étudiant doit aussi apprendre à chercher l'information pertinente.

Connaissances préalables recommandées : connaissance algorithmique, langage de programmation orienté web.

Contenu de la matière :

1. Introduction
2. Éléments clés en RI
 - Type de recherche
 - Fonctionnement de la RI par requête
 - Type d'indexation
3. Modèle de RI
 - Modèle Booléen
 - Modèle Booléen Pondéré
 - Modèle Vectoriel
 - Modèle Probabiliste
 - Classique
 - Réseau d'inférence
 - Modèle de Langue
4. Systèmes de recherche d'information (SRI) – Architecture
5. Recherche d'information sur le Web : calcul de score de notoriété, spécificités du Web
6. Évaluation de SRI
7. Conclusion

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

Références

1. W. Bruce Croft, Donald Metzler, Trevor Strohman. 29. Search Engines: Information Retrieval in Practice, Pearson Education
2. Charles L. A. Clarke, Gordon V. Cormack, and Stefan Büttcher Information Retrieval: Implementing and Evaluating Search Engines. MIT Press, 2010
3. Christopher D. Manning, Prabhakar Raghavan and Hinrich, 28. Introduction to Information Retrieval, Cambridge University Press.
4. R. Baeza-Yates, B. Ribeiro-Neto, Modern Information Retrieval, Addison-Wesley, 1999.
5. G. Salton, M.J. McGill, Introduction to Modern Information Retrieval, McGraw-Hill, 1983 (épuisé, disponible à la bibliothèque)
6. G. Salton, Automatic Text Processing, Addison-Wesley, 1989 (épuisé, disponible à la bibliothèque)
7. C.J. van Rijsbergen, Information Retrieval, Butterworths, 1979. Disponible en version électronique (<http://www.iro.umontreal.ca/~nie/IR-book/Preface.html>)

Semestre: 6

Unité d'enseignement : UEF62

Matière : Fondements de l'intelligence artificielle

Crédits : 5

Coefficient : 3

Objectifs de l'enseignement : L'objectif de ce module est de doter l'étudiant par les concepts fondamentaux de l'intelligence artificielle. L'intérêt de ce module portera sur l'apport de l'IA en matière de résolution de problème qui n'est plus dans le contexte de l'informatique de calcul mais plutôt des problèmes qui nécessitent une approche exploitant le principe du raisonnement.

Connaissances préalables recommandées : logique mathématique, fondement algorithmique

Contenu de la matière :

Chapitre 1 Introduction générale à l'intelligence artificielle

- Qu'est-ce que l'intelligence artificielle (IA) ?
- Comparaison entre apprentissage naturel et apprentissage artificiel
- Place de l'IA dans les sciences informatiques et ses domaines d'application

Chapitre 2 Présentation du Machine Learning

- Définition et objectifs du Machine Learning
- Les principaux types d'apprentissage : supervisé, non-supervisé, par renforcement
- Principes de l'apprentissage par exploration
- Notions d'optimisation dans le cadre de l'apprentissage
- Apprentissage par approximation et interpolation

Chapitre 3 L'environnement méthodologique de l'apprentissage

- L'espace de données d'apprentissage : entrées, sorties, features
- Phases du processus d'apprentissage : entraînement et test
- Stratégies de validation des modèles : validation croisée, hold-out, etc.

Chapitre 4 Algorithmes d'apprentissage supervisé

- Régression linéaire et modèles paramétriques
- Réseaux de neurones artificiels (perceptron, MLP)
- Méthode des k plus proches voisins (k-NN)
- Machines à vecteurs de support (SVM)

Chapitre 5 Le Clustering

- Pourquoi partitionner les données : objectifs et cas d'usage
- Évaluer la qualité d'un clustering (indices internes et externes)
- Clustering hiérarchique (agglomératif/divisif)
- Algorithme des k-moyennes (k-means)
- Clustering basé sur les modèles : mélange gaussien
- Clustering par densité : DBSCAN
- Clustering spectral : principes et intérêt

Chapitre 6 Deep Learning et réseaux avancés

1. Introduction au Deep Learning
2. Applications courantes : reconnaissance visuelle, traitement du langage naturel
3. Limites : données massives, coût de calcul, boîte noire

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

Références

1. Manuel d'intelligence artificielle, Louis Frécon et OkbaKazar, édition PPUR, ISBN:978-2-88074-819-7, 29
2. Ganascia, Jean-Gabriel. L'intelligence artificielle. Flammarion, 1993.
3. I. Bratko, Programmation en Prolog pour l'intelligence artificielle, 21
4. J.M. Alliot et T.Schiex, Intelligence Artificielle et Informatique Theorique, Cepadues Editions, 1993.
5. N. Nilsson, ArtificialIntelligence: A New Synthesis, Morgan Kaufmann, 1998.
6. S. Russell, P. Norvig, Artificial Intelligence: A Modern Approach, 2nd edition, 22

Semestre: 6

Unité d'enseignement : UEM61

Matière : Projet

Crédits : 6

Coefficient : 4

Objectifs de l'enseignement : l'objectif de cette matière est de confier à un groupe d'étudiants le soin d'analyser, de concevoir et d'implémenter une application informatique.

Connaissances préalables recommandées : Cours de la licence SI

Contenu :

Le projet de licence porte sur une ou plusieurs thématiques abordées dans le cursus de la licence SI. Il est encadré par un enseignant du département.

Il peut être effectué dans une entreprise (stage) ou au niveau du département.

Evaluation :

L'évaluation se fera par un jury d'examen composé de trois (03) enseignants : Le président, l'examineur et l'encadreur.

L'examen portera sur :

- Le mémoire (ou rapport de stage) : 07 points.
- Le logiciel : 07 points.
- Les réponses aux questions : 06 points

La note finale est égale à la moyenne des notes attribuées par chacun des membres du jury d'examen.

Il n'y a pas de soutenance. La consultation porte sur le manuscrit et le logiciel (cotés conception et réalisation).

Le mémoire (ou rapport de stage) comporte l'essentiel du travail et ne doit pas dépasser 30 pages.

Note = Rapport (35%) + Logiciel (35%) + Réponses (30%)

Semestre: 6

Unité d'enseignement : UEM61

Matière : Intelligence métier (BI)

Crédits :2

Coefficient : 1

Objectifs de l'enseignement : l'objectif de cette matière est d'apporter des connaissances à l'étudiant pour :

1. Comprendre le business intelligence
2. Déterminer les différentes étapes dans la mise en place d'un projet B.I
3. Examiner le marché de la BI et les outils leaders

Connaissances préalables recommandées : maîtrise de base de données relationnelles

Contenu de la matière :

1. BI vs ERP
2. Architecture classique d'une solution BI
3. ETL et Data
4. Modèles multi-dimensionnels et OLAP
5. Data Warehouse Modeling
6. Data Warehouse architecture
7. BI et Data Mining
8. KPI, Dashboard, Scorecards et Cockpit
9. Real-time BI

Mode d'évaluation : examen final (100 %)

Références

1. Alphonse Carlier Business Intelligence et Management, AfnorÉditions, 2013, (ISBN978-2-12-465429-1); EAN : 9782124654291
2. Jean-Marie Gouarné, Le Projet décisionnel - Enjeux, Modèles, Architectures du Data Warehouse [archive], Eyrolles, 1997, (ISBN978-2-212-05012-7)
3. Alain Garnier, L'Information non structurée dans l'entreprise - Usages et Outils, Hermes - Lavoisier, 27, (ISBN978-2-7462-1605-1)
4. R. Kimball, L. Reeves, M. Ross, W. Thornthwaite, Le Data Warehouse : Guide de conduite de projet, Eyrolles, 25, (ISBN978-2-212-116-7)
5. Alain Fernandez, Les Nouveaux Tableaux de bord des managers, Le Projet Business Intelligence clés en main, Eyrolles, 6e édition, 2013. (ISBN978-2-212-55647-6) présentationéditeur [archive]
6. Roland et Patrick Mosimann, Meg Dussault, The Performance Manager Faire de la performance le quotidien de chacun [archive], CognosPress, 27, (ISBN978-0-9730124-4-6)
7. James Taylor, Decision Management System [archive], IBM Press, Pearson Education

Semestre:6

Unité d'enseignement : UET61

Matière : Déontologie de l'informatique

Crédits : 2

Coefficient : 1

Objectifs de l'enseignement :

L'utilisation d'un outil informatique peut être plus difficile dans des situations dominées par un conflit éthique. Il existe un risque important que des systèmes informatiques coûteux mais vitaux soient abandonnés en raison de scandales et de désaccords. Il est devenu nécessaire, pour l'entreprise et pour la direction informatique, de se prémunir contre les risques liés à des usages juridiquement non-conformes des systèmes informatiques. Pour ce faire, plusieurs solutions existent, notamment la définition de règles déontologiques acceptables par tous. Ainsi donc ce cours s'intéresse essentiellement à la déontologie et à l'éthique dans le monde informatique.

Connaissances préalables recommandées : Connaissances de base en informatique acquises dans les trois premiers semestres.

Contenu de la matière :

Chapitre 1 - Ethique & Déontologie – Quelques notions

1. Morale
2. Valeurs
3. Préjugés
4. Ethique
5. Déontologie

Chapitre 2 - Théories de l'éthique

1. Autorités (Religieuse, Sociale, Scientifique)
2. Théories Intuitionnistes (Intuitionnisme)
3. Théories Egoïste
4. Théories Conséquentialistes
5. Théories Déontologiques

Chapitre 3- Déontologie et éthique informatique

1. Quelques codes d'éthique et de conduite professionnelles informatiques
 - Les 10 commandements de l'éthique informatique
 - Code d'éthique et de conduite professionnelle de l'international federation for information processing

Chapitre 4 - Ethique en recherche et développement

1. Plagiat & self-plagiat.
2. Falsification des résultats

Chapitre 5 - Corruption et déontologie de travail.

1. Concepts de la corruption
2. Types de corruption
3. Lutte contre la corruption par les organismes et les organisations locales et internationales
4. Méthodes de traitement et moyens de lutter contre le phénomène de la corruption

Mode d'évaluation : Examen (100%)

Références :

1. Déontologie des usages des Systèmes d'information – Principes fondamentaux, CIGREF, Janvier 26
2. Ethique et Déontologie, A. GAHMOUSSE
3. Code d'éthique et de conduite professionnelle de l'international federation for information processing», 132-133, 2022
4. The Ten Commandments of computer ethics have defined by the Computer Ethics Institut