

Programme 3^{ème} Année Informatique

Semestre 5	CM	TD	TP	Crédits
<u>UEI3 (Fondamentale)</u>	-	-	-	18
Systèmes d'Exploitation 2	3h	1h30	1h30	6
Compilation	3h	1h30	1h30	6
Réseaux	3h	1h30	-	6
<u>UEI4 (Fondamentale)</u>	-	-	-	12
Programmation Logique	1h30	1h30	-	4
Théorie des Graphes	1h30	1h30	-	4
+ 1 option au choix parmi : - Infographie - Ingénierie des connaissances - Cryptographie - Développement des interfaces graphiques - ou autres options à proposer par les établissements	1h30		-	4
Total (24h , 30 crédits)	13h30	7h30	3h	30

Semestre 6	CM	TD	TP	Crédits
<u>UEI15 (Fondamentale)</u>	-	-	-	10
2 options au choix parmi une liste proposée par les établissements (avec recommandation que les deux options choisies soient liées au projet)	1h30	-	-	5
	1h30	-	-	5
<u>UEI16 :Projet</u>	-	-	-	20
Total (3h présentiel + Projet , 30 crédits)	3h	-	-	30

3^{ème} Année L3 (Licence d'Informatique)

Semestre 5

UEI13 (Fondamentale) 18 crédits

- Systèmes d'Exploitation 2
- Compilation

Objectif:

Introduction au problème de la compilation :

- du texte-source au code assembleur en passant par l'arbre de syntaxe abstraite

- sur la base d'un petit compilateur réalisé en C, en trois versions de complexité croissante. L'assembleur visé est celui du cours d'Architecture. Technique classique d'analyse syntaxique : Lex et Yacc. Le but du cours est de montrer le rôle de la pile dans la compilation des fonctions sur le modèle de C.

Programme :

- Introduction, position du problème, plan du cours.
- Syntaxe abstraite et interprétation de l'Assembleur du cours d'architecture.
- Génération de code pour les expressions arithmétiques et pour les structures de contrôle (Assembleur cours d'Architecture).
- Analyse lexicale – Lex.
- Analyse syntaxique – Yacc.
- Application : Un lexeur-parseur pour Assembleur cours d'Architecture.
- Blocs : principe (Assembleur cours d'Architecture).
- Blocs : réalisation (Assembleur cours d'Architecture).
- Fonctions : principe (Assembleur cours d'Architecture).
- Fonctions : réalisation (Assembleur cours d'Architecture).
- Procédures (Assembleur cours d'Architecture).

Références bibliographiques :

- Aho, Sethi, Ullman : Compilers (Addison-Wesley) Trad. française chez InterÉditions
- Christopher Fraser and David Hanson. A Retargetable C Compiler : Design and Implementation. Benjamin/Cumming, 1995

• Réseaux

Objectif :

Ce module est une introduction au monde des réseaux informatiques. Il constitue un noyau de base des connaissances " réseaux " dont la compréhension est essentielle, car tous les concepts présentés sont utilisés dans les réseaux actuels et à venir. Il a comme objectifs :

- la compréhension des concepts fondamentaux utilisés en réseaux : structuration d'une architecture de communication en couches, notion de protocole, principaux mécanismes de communication tels que : contrôle d'erreur, contrôle de flux, contrôle de congestion, commutation, mode de communication, routage, adressage ;
- l'initiation à des protocoles courants, notamment Ethernet, IP, TCP ;
- la familiarisation de l'étudiant avec le vocabulaire " réseau ".

Nous insisterons ainsi sur les infrastructures de transport de l'information, utilisées par les applications de l'Internet. À la fin du semestre, des séances de TD sur machine permettent de mettre en pratique les notions vues et d'effectuer une synthèse des connaissances acquises.

Programme :

- Qu'est-ce qu'un réseau ? Définitions, classifications, architectures
- Transmission physique de l'information : traitement du signal, signal numérique/analogique, codage, circuit de données, modems
- Fiabilisation de la transmission : contrôle d'erreur, contrôle de flux, illustration avec les protocoles HDLC et PPP
- Réseaux locaux : plan de câblage, topologie, méthodes d'accès au support de communication, illustration avec Ethernet et Token Ring
- Réseaux grande distance : techniques de commutation, adressage, routage, contrôle de congestion, illustration avec des réseaux d'opérateurs (X.25, Relais de Trames ou Frame Relay, ATM)

- Interconnexion de réseaux : le protocole IP, les équipements d'interconnexion (routeurs, commutateurs ou switches, ponts, hubs).
- Communications de bout en bout : fiabilisation et optimisation du transport de l'information, numéros de ports, multiplexage, segmentation, contrôle d'erreur, contrôle de flux, illustration avec les protocoles TCP et UDP
- Commandes Unix pour la configuration et l'administration d'un réseau
- Installation et configuration d'un serveur web et d'un proxy
- Fonctionnement du protocole IP et adressage IP ; interconnexion de réseaux : notions de passerelles
- Fonctionnalités de la couche Transport, en mode connecté et en mode non connecté ; techniques d'adressage au niveau transport
- Fonctionnement des protocoles UDP et TCP ; les sockets
- Exemples d'applications : http, mail ; accès au réseau pour les utilisateurs ; serveurs DNS

Références bibliographiques

- Comer, D. Ed.: InterEditions. *TCP/IP: Architecture, Protocoles, Applications*.
- Rolin, P. Ed. : Hermès. *Réseaux locaux, normes et protocoles*
- Tanenbaum, A.. Ed.: InterEditions. *Réseaux: Architectures, Protocoles, Applications*.

UEI14 (Fondamentale) 12 crédits

• Programmation Logique

- Introduction à la programmation logique : langage PROLOG
- Principales caractéristiques de ce type de programmation
- Syntaxe et structures de données – opérateur de coupure
- Sémantique des programmes PROLOG
- Le problème de la négation en PROLOG : l'hypothèse du monde clos et la négation par échec.
- Utilisation de la méthode de résolution dans l'implantation machine de ce type de langage.

Références bibliographiques :

- Chazarain, Programmer avec SCHEME . De la pratique à la théorie. Thomson International, 1996.
- Hoogger. Programmer en logique. Masson, 1987
- Weis & Leroy. Le langage CAML. Interéditions, 1993.

• Théorie des Graphes

Objectifs (compétences visées) : Appréhender les algorithmes des graphes utilisés dans les réseaux informatiques, dans les problèmes de calcul de coût minimal, dans la recherche du meilleur chemin et dans les méthodes d'ordonnancement (Gestion des projets,...)

Programme :

- **Notions fondamentales de la théorie des graphes**
 - Définitions d'un graphes et différentes représentations
 - Applications multi-graphes
 - Applications multivoque, degré, demi degré, cycles, cocycles, connexité...
- **Les nombres fondamentaux de la théories des graphes**
 - Nombres de stabilités

- Noyau, fonction ordinale, fonction de Grundy
- Nombre chromatique, clique, théorème de Koenig
- **Graphes particuliers**
 - Graphes planaires, Graphe dual
 - Graphes aux arêtes, graphes aux arcs
- **Arbres et Arborescence**
 - Construction d'un arbre
 - Construction d'une forêt
 - Algorithme de Kruskall (cycles et cocycles)
 - Algorithme de Sollin
- **Problèmes de flots**
 - Définitions
 - Cycles élémentaires et flots élémentaires
 - Problème du flot maximal dans un réseau de transport
 - Graphe d'écart
 - Algorithme de recherche du flot maximal (Ford-Fulkerson)
- **Problèmes du plus court chemin**
 - Introduction au problème du plus court chemin
 - Algorithme de Dantzig
 - Algorithme de Ford
 - Algorithme de Dijkstra
- **Méthodes d'ordonnement**
 - Diagramme de Gantt
 - Méthode PERT
 - Méthode MPM

● Infographie

- Techniques de base en infographie
- Systèmes graphiques
- Communication graphique
- Modélisation Géométrique
- Utilisation des outils (API, Outils standards)
- Animation
- Visualisation
- Réalité virtuelle
- Vision Machine

● Ingénierie des connaissances

Objectifs : la conception d'un prototype de système de représentation de connaissances

- Acquisition des connaissances
 - Apprentissage numérique
 - Apprentissage symbolique
- Représentation des connaissances
 - Les différents types de connaissances
 - Connaissances factuelles
 - Connaissances ontologiques
 - Connaissances assertionnelles
 - Les formalismes de représentation des connaissances

- Frame
- Réseaux sémantiques
- Graphes conceptuels
- Logique
- Les différents langages de représentation des connaissances
 - KIF
 - XML
 - RDF et RDFS
 - La logique de description
- Les systèmes de représentation des connaissances
 - KL-ONE
 - CLASSIC
 - LOOMS
- Des systèmes experts aux systèmes à base de connaissances
- Modélisation des connaissances
- Conception et réalisation d'ontologies en ingénierie des connaissances
- Raisonnement à partir de cas
- Gestion des connaissances

Références Bibliographiques

J. Charlet, M. Zacklad, G. Kassel, D. Bourigault. « Ingénierie des connaissances : Evolutions récentes et nouveaux défis ». Editions Eyrolles. France, 2000.

• Cryptographie

- Initiation aux concepts fondamentaux et aux méthodes de la cryptographie contemporaine.
- Principes de la cryptographie à clef publique, ses avantages par rapport à la cryptographie classique qui oblige à garder secrètes les clefs de chiffrement et de déchiffrement.
- Etude de quelques cryptosystèmes tel que le célèbre RSA, et protocoles d'échanges de clefs, d'authentification, de chiffrement.
- Génération de nombres pseudo-aléatoires.

• Développement des Interfaces Graphiques

- Fonctions d'une interface
- Styles d'interactions et types d'interfaces
- Conception des IHM (modèle en cascade, prototypage)
- Outils de construction des IHM
- Le système X-Windows (protocole X11, bibliothèque Xlib, bibliothèque Xt, les boites à outils)
- Programmation d'interfaces (TCL/TK, Motif et Java)
- Règles d'interfaçage pour les IUG (Interfaces Utilisateurs Graphiques)
- Evaluation des Interfaces

Références bibliographiques :

- M. Campione, K. Walrath. The Java tutorial, (second edition) 1998.
- A. Nye et T.O'Reilly. The definite Guide to X-Windows System, (Motif Edition) O'Reilly&Associates.
- Welch Brent balding. Practical Programming in TCL/TK (second edition), Prentice hall, 1997.

UEI15 (Fondamentale) 10 crédits

- 2 options au choix parmi une liste établie par l'établissement

UEI15 (Fondamentale) 20 crédits

- **Projet**