

**DESS**  
**COMPÉTENCE**  
**COMPLÉMENTAIRE**  
**EN INFORMATIQUE**

Formation Continue

**- Objectifs de la formation :**

Donner en un an un bon niveau de compétence en Informatique à des spécialistes d'autres disciplines scientifiques.

Acquérir d'une part un savoir-faire en conception et construction d'applications dans les secteurs d'activité relevant de la compétence initiale, et d'autre part la maîtrise des fonctions des systèmes et des réseaux informatiques.

**- Conditions d'accès :**

La formation est ouverte aux personnes titulaires d'une formation en Informatique de type Bac+2 au moins et justifiant d'une expérience professionnelle significative, ou être titulaires d'un diplôme scientifique de niveau Bac+4 dans une discipline distincte de l'Informatique. Le nombre maximum d'étudiants pouvant être admis chaque année dans cette formation est de 20.

Le dossier de candidature est à demander au DEPULP.

**- Inscription :**

DEPULP

Arnaud WESTERMEYER

21 Bd du Maréchal Lefebvre

67100 STRASBOURG – 03 88 40 75 55

**- Organisation :**

Trois périodes : enseignement, projet, stage. Il est compté 450 heures pour l'enseignement et le projet, réparties sur 11 semaines, de janvier à décembre.

Le projet d'un volume horaire de 100h, correspond à l'analyse et à la réalisation d'un environnement matériel ou logiciel. Un rapport écrit doit être produit.

Le stage d'une durée d'au moins 4 mois permet aux étudiants non salariés de connaître la vie d'une entreprise, de travailler sur un problème réel, de s'insérer dans une équipe et de participer à la conception et au développement d'un projet. Le stage doit se concrétiser par la réalisation d'une partie de ce projet, sous la forme d'un matériel ou d'un logiciel. Ce travail donne lieu à un rapport écrit et à une soutenance orale.

**- Débouchés :**

Conception et réalisation d'applications relevant de la compétence initiale. Métiers à l'interface entre les informaticiens et les spécialistes des autres branches.

**- Responsable formation continue :**

Pierre Tellier

Le programme ci-dessous est celui écrit lors de la demande de renouvellement de l'habilitation du diplôme. Son contenu est sujet à des modifications mineures, en fonction de l'évolution du marché et des orientations proposées par les enseignants, en particulier ceux issus du secteur privé.

|                                                                            |
|----------------------------------------------------------------------------|
| <b>Enseignement</b>                                                        |
| <b>ENVIRONNEMENTS</b><br>Cours : 15h - TP : 15h                            |
| <b>ALGORITHME ET PROGRAMMATION</b><br>Cours : 50h - TP : 30h               |
| <b>PROGRAMMATION OBJET</b><br>Cours : 30h - TP : 50h                       |
| <b>SYSTÈMES D'EXPLOITATION</b><br>Cours : 50h - TP : 30h                   |
| <b>RÉSEAUX</b><br>Cours : 50h - TP : 30h                                   |
| <b>BASES DE DONNÉES</b><br>Cours : 20h - TP : 20h                          |
| <b>GÉNIE LOGICIEL ET "PROJET</b><br>Cours : 20h - TP : 30h - Projet : 100h |
| <b>STAGE EN ENTREPRISE (POUR NON SALARIES)</b><br>4 mois minimum           |

## **ENVIRONNEMENTS**

**ENSEIGNANT** : Intervenant externe

### **CONTENU DU COURS :**

Installation du matériel et des systèmes d'exploitation (Linux et NT). Installation des principaux outils et mise en place de l'environnement de travail : logiciels, connexion au réseau, messagerie, imprimante etc. Fonctionnement et utilisation des environnements graphiques (X11, NT). Gestion des fichiers, des répertoires, utilisation d'un éditeur de texte, configuration de l'environnement (fichiers ressources et profils utilisateurs). Outils bureautique et utilisation avancée : liaison, fusion, échange de données.

## **PROGRAMMATION**

**ENSEIGNANT** : P. Tellier, A. Sartout, J.M. Gallonne (ULP)

### **CONTENU DU COURS :**

Éléments de bases de l'algorithmique : séquence, test et itération. Variables et types. Expressions. Entrées-sorties simples. Tableaux et structures de données. Pointeurs.

Fichiers. Programmation modulaire. Bibliothèques standards. Création de bibliothèques réutilisables. Préprocesseur, gestionnaire de projets. Programmation d'interfaces graphiques. Programmation CGI.

Travaux pratiques : Compilateurs C et C++ : GNU gcc/g++, Visual Basic, Microsoft Visual C++ , HTML.

### **RÉFÉRENCES :**

AHO A.V., & al., Data Structures and Algorithms, Addison-Wesley, 1983.

KERNIGHAN B.W., RICHIE D.M., Langage C norme Ansi 2ème édition, Dunod, 1997.

## ***PROGRAMMATION ORIENTÉE OBJET***

**ENSEIGNANT :** D. Wagner (ULP)

### **CONTENU DU COURS :**

Notions de base de la programmation orientée objet : encapsulation et abstraction. Introduction à l'analyse et à la conception orientée objet. Notions d'objet et de classe. Structure d'un programme orienté objet. Classe : syntaxe, description et utilisation. Instances. Spécialisation et héritage. Typage statique et dynamique. Entrées-sorties. Bibliothèques et classes standards. Fenêtrage et programmation événementielle. Applications/Applets. Amélioration de sites sur le World-Wide-Web. Programmation réseau : utilisation de ressources distantes, accès aux bases de données. Développement de composants réutilisables. Utilisation du multimédia.

Travaux pratiques : JDK et bibliothèques associées.

### **RÉFÉRENCES :**

FLANAGAN, D., Java in a nutshell, O'Reilly France, 2000

## ***SYSTÈMES D'EXPLOITATION***

**ENSEIGNANT :** F. Soulier (Aztec Radiomedia), P. Thomas (Bull), C. Haessig, S. Thery (IRCAD)

### **CONTENU DU COURS :**

Composants des systèmes informatiques. Éléments d'architecture des ordinateurs : unité centrale, mémoire, bus, périphériques. Principe et fonctionnement d'un système d'exploitation : gestion des ressources et besoins des utilisateurs. Structure des systèmes d'exploitation. Processus et threads. Gestion de la mémoire. Entrées-sorties et périphériques. Systèmes de fichiers. Protection des ressources. Interpréteur de commandes. Appels système. Outils et utilitaires système, programmation du shell pour les tâches d'administration. Outils de communication réseau : transfert de fichiers (ftp), connexion à distance (telnet, rlogin). Installation et configuration matérielles. Installation du système (partitionnement des disques, installation et gestion des périphériques) et des applications. Ajout d'utilisateurs. Mise en place d'une politique d'accès aux ressources : unités de stockage (disques, sauvegardes, cédéroms), imprimantes. Administration système : gestion et configuration d'une station ou poste de travail, d'un serveur. Intégration au réseau : configuration des services NIS, NFS, DNS. Messagerie : serveur (smtp, pop, imap,...) et clients. Configuration de terminaux alphanumériques (connexion

série : getty) et graphiques (xdm, Wincenter). Aspects de la sécurité. Intranet par l'intermédiaire de http. Études de cas : des besoins aux solutions.

Travaux pratiques : Microsoft Windows NT workstation/NT server et Linux. Terminaux VT100 et terminaux X. Logiciels Samba, sendmail, popper, imapd, WinCenter, Apache httpd,...

## **RÉFÉRENCES :**

TANENBAUN A., Systèmes d'exploitation, DUNOD

SILBERSCHATZ A., Principes des systèmes d'exploitation, Addison & Wesley

WELSCH M., Le système Linux, O'Reilly

LERDORF R., PHP Pocket Reference, O'Reilly & Associates, Jan 2000, ISBN 1-56592-769-9, 114p.

CASTAGNETTO J., Professional PHP Programming, Wrox Press Inc, Dec 1999, ISBN 1-86100-296-3, 909 pages

ATKINSON L., Core PHP Programming, Prentice Hall, April 1999, ISBN 0-13-020787-X, 568 pages

## **RÉSEAUX**

**ENSEIGNANTS :** P. Babocsay, T. Joly (Alcatel), Philippe WAGNER (Lilly France)

## **CONTENU DU COURS :**

Utilisation et applications des réseaux. Les principaux composants des réseaux informatiques. Normes. Réseaux locaux (LAN) et étendus (WAN). Modèle OSI, notion de protocole et de service. Protocoles TCP/IP (ARP, ICMP, IGMP, UDP, TCP, IP, IPv6 (routage, adressage, subnetting...)). Topologie des réseaux : bus, anneau et étoile. Méthode d'accès : jetons. Médias : câblage (paires torsadées, fibre optiques...). Interconnexion : répéteurs, ponts, passerelles, concentrateurs, commutateurs, routeurs, firewalls. Méthodologies et outils d'analyse et de diagnostic de réseaux. Architectures de gestion TCP/IP (SNMP). Administration de parc de machines. Accès RTC, modems. Applications: vidéo ou audio conférence, espace de travail partagé à distance. Illustration avec l'infrastructure du Département d'Informatique de l'ULP (architecture réseau, armoires, câblages).

Travaux pratiques : Voir module « Systèmes d'exploitation ». Routeurs CISCO, concentrateurs, analyseur réseau, modems.

## **RÉFÉRENCES :**

TANENBAUM A., Computer networks (F : Architectures, réseaux et protocoles)

COMER D., TCP/IP Architecture, Protocoles, Applications

COMER D., Internetworking with TCP/IP

NAIK D.C., Standards et protocoles de l'internet, Microsoft Press

PUJOLLE G., Les Réseaux, 3ème édition, Eyrolles, 2000, ISBN: 2-212-09119-2

MONTAGNIER J.L., Pratique des Réseaux d'entreprise, Eyrolles

## **BASES DE DONNÉES**

**ENSEIGNANTS :** S. Brandel (ULP), D. Martin (Ventana Medical Systems)

**CONTENU DU COURS :**

Introduction à la modélisation : schéma entités-associations. Notion d'algèbre relationnelle. Utilisation des dépendances fonctionnelles pour la normalisation. Opérations sur les relations : sélection, union, projection, jointure, intersection. Le modèle relationnel : relations, attributs, clés primaires et étrangères, contrainte d'intégrité. Le langage SQL : définition et manipulation des données, interrogation. Éléments d'administration : conception physique, contrôle d'accès, sauvegardes, reprise à chaud (warm start). Transactions et aspect client/serveur. Bases de données actives.

Travaux pratiques : Oracle.

**GÉNIE LOGICIEL**

**ENSEIGNANT :** N. Louis (Business Design Consulting)

**CONTENU DU COURS :**

Introduction à la planification : tâches et ressources, diagrammes PERT, tableaux de Gantt. Modélisation orientée objet : identification des classes, des attributs et des associations, utilisation des hiérarchies d'héritage. Conventions des diagrammes de la méthode UML. Outils pour le génie logiciel : constructeur d'application, générateur d'interface graphique, gestion de version, travail coopératif, documentation automatique. Outils de vérification du code.

Travaux pratiques : Rational Rose, Environnements de développement Microsoft (Visual C++, J++), Outils GNU (make, sccs, rcs, cvs), Sun JDK javadoc, Insure/Purify.

**COACHING (à confirmer)**

**ENSEIGNANT :** C. MEGEL (Actimage)

**CONTRÔLE DES CONNAISSANCES**

L'enseignement du DESS CCI est décomposé en trois parties :

Une partie théorique comprenant les enseignements décrits dans la plaquette de présentation du DESS CCI, dont les coefficients sont tous identiques. Ils donnent lieu à un contrôle organisé par le responsable de chaque enseignement, sous forme d'épreuve écrite non anonyme, orale ou pratique. Cette partie donne lieu à une première session d'évaluation notée sur 20. Une deuxième session est organisée pour les candidats ayant échoué à la première.

Un projet faisant l'objet d'un travail personnel important (100h) et donnant lieu à une réalisation concrète. Ce travail est évalué par l'enseignant responsable, et noté sur 20.

Un stage en entreprise donnant lieu à un rapport écrit et à une soutenance orale. La note de la partie stage, sur 20, est la moyenne des notes de rapport, de présentation orale et de travail, délivrées par le jury de stage comprenant enseignants et encadreurs du stage.

## **Délivrance du diplôme**

Ces trois notes entrent dans le calcul d'une note globale. La note de la partie théorique intervient avec un coefficient 7, celle du projet avec un coefficient 2 et celle du stage en entreprise avec un coefficient 3. Le candidat est admis si chacune d'elles est supérieure ou égale à la moyenne. Une mention est délivrée en fonction de la note globale.

## **Jury**

Pierre Tellier, MCF ULP

Sylvain Brandel, MCF ULP

Michel Kretz, MCF ULP