

Licence mention Informatique

Parcours Informatique : Fondements et Applications

Objectifs

Les objectifs de la licence d'informatique sont l'acquisition des compétences fondamentales, méthodes et savoir-faire techniques représentatifs des différentes tâches de la discipline informatique. Cette formation couvre l'ensemble de la discipline informatique : fondements, architecture et matériel, réseaux, systèmes, méthodes et technologies logicielles, applications informatiques, systèmes d'information.

Le parcours IFA est le parcours généraliste sur 3 ans de la licence informatique. En plus des compétences informatique, ce parcours initie les étudiants aux bases de l'approche scientifique et à la capacité d'abstraction, ainsi qu'à la démarche de recherche et d'analyse. La formation vise également à donner aux étudiants des compétences transversales et scientifiques générales.

Le parcours IFA est accessible en première année à partir du portail [ISI](#) (menant aux mentions Informatique et [Sciences pour l'Ingénieur](#)) et accessoirement du portail [MPMEI](#) (menant aux mentions [Mathématiques](#), [Physique](#), Maths-Eco et Informatique).



Compétences acquises

- > Analyse, modélisation d'un problème, conception et mise en œuvre d'une solution informatique adaptée en choisissant des outils adéquats.
- > Conception d'IHM, d'applications Web, de systèmes d'information.
- > Utilisation de différents langages de programmation et systèmes d'exploitation.
- > Apprentissage des architectures matériels et des réseaux.
- > Compétences scientifiques non informatique (mathématiques, physiques et/ou électronique) en première année.

La formation comprend un stage disciplinaire obligatoire de 8 semaines minimum en fin de 3ème année. Ce stage peut également être effectué dans une université étrangère en partenariat avec le département d'informatique de l'UBO.

Conditions d'accès

Baccalauréat ou équivalent, ou sur validation des acquis de l'expérience (VAE). L'admission directe en deuxième et troisième année se fait sur dossier.

Poursuites d'études

Accès en Master Professionnel ou Master Recherche à dominante informatique, en écoles d'ingénieurs.

Insertion professionnelle

Le diplômé peut exercer dans des sociétés de service informatique ou dans le service informatique d'une entreprise (quelque soit le secteur d'activité de celle-ci).

Il peut exercer les emplois suivants :

- > Développeur informatique (systèmes d'information, applications Web, ...)
- > Assistant fonctionnel/Technicien support
- > Administrateur systèmes et réseaux
- > Administrateur de bases de données

Infos pratiques

Faculté des Sciences et Techniques à Brest
Ouvert en stage

Contacts

Responsable pédagogique

Lallali Mounir
Mounir.Lallali@univ-brest.fr
Tel. 0298018139

Responsable Secrétariat pédagogique

Secrétariat du département informatique
dept-info@univ-brest.fr
Tel. 02 98 01 62 06 / 80 11

Programme

Licence 2ème année

Semestre 3

Bases de Données Relationnelles	26h
Programmation C	48h
Bloc transversal S3	
- Anglais	16h
- Culture scientifique	16h
- Compétences numériques	
Projet de conception objet	54h
Réseau : modèle OSI	18h
Développement Web côté client	20h
Langages formels et automates	36h

Semestre 4

Algorithmique fondamentale, Graphes et Test	56h
Architecture et Systèmes 1	48h
Logique	20h
Programmation fonctionnelle	54h
Java 1	20h
Developpement Web côté serveur	26h
Bloc transversal S4 I	
- Anglais	16h
- Option transversale S4	11h
- Culture scientifique	16h
- Sea-EU / Sport, culture, engagement étudiant	
- Preparation à l'Expérience Professionnelle en Synergie en Informatique	16h

Licence 3ème année

Semestre 5

Programmation C avancée	54h
Programmation logique et Intelligence artificielle	54h
Ingénierie des systèmes d'information	54h
Java et Conception d'Applications	
- Java 2	36h
- Conception d'applications	20h
Bloc transversal S5 - IFA	
- Anglais S5	16h
- Communication S5	12h
- Option ouverture Master (1 au choix)	
- Administration système	22h
- Objets connectés et robotique	22h
- Sécurité	22h

Semestre 6

Projet applicatif	70h
Algorithmique avancée	48h
Architectures et Systèmes 2	
- Architectures 2	28h
- Systèmes 2	28h
Réseaux IP	
- Réseaux IP : programmation des réseaux	28h
- Réseaux IP : RPC	16h
- Réseaux IP : IHM	27h
Bloc transversal S6 IFA-CDA	
- Stage (8 semaines)	280h
- Anglais S6	16h
- Communication S6	12h

Dernière mise à jour le 01 mars 2022

Bases de Données Relationnelles

Présentation

Cet enseignement couvre le domaine de la conception de bases de données relationnelles et de l'interrogation avec le langage SQL.

- > Conception de bases de donnée avec UML
- > Passage du modèle conceptuel de données UML au modèle physique
- > Mise en place d'une base de données relationnelle avec SQL
- > Interrogation d'une base de données avec SQL
 - > Projection
 - > Sélection
 - > Jointures (Jointure Naturelle et Jointure Externe)

3 crédits ECTS

Volume horaire

Cours Magistral : 8h

Travaux Pratiques : 10h

Travaux Dirigés : 8h

Compétences visées

- > savoir modéliser des données avec le diagramme de classes UML en vue de mettre en œuvre une base de données relationnelle.
- > savoir utiliser des requêtes simples d'interrogation de BDR avec le langage SQL

Descriptif

- > Définitions ; BD et SGBD.
- > Conception d'une base de données ; modèle entités-associations ; Merise ; UML.
- > Bases relationnelles ; passage du modèle conceptuel au modèle logique. Langage SQL (traduction des opérateurs de l'algèbre relationnelle dont la division, aspect calculatoire) ; SQL sous Oracle.
- > Interfaçage et diffusion d'une base de données ; PHP ; formulaires et pages web.
- > Niveau physique et optimisations ; organisation d'un index de table relationnelle.

Bibliographie

- > [SQL](#) / Frédéric Brouard, Rudi Bruchez, Christian Soutou. Pearson

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	120	2/3	
UE	CC	Travaux Pratiques		1/3	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	120	1/1	

Programmation C

Présentation

Le but de ce cours est d'obtenir un socle de base en langage C, pré-requis dans plusieurs autres cours de la formation. À l'issue de ce cours, les étudiant-e-s devraient notamment être capable :

- > de faire un programme simple en C (écriture du programme, compilation puis exécution)
- > d'utiliser un certain nombre de fonctions très classiques de la bibliothèque standard C
- > de comprendre la documentation de celle-ci
- > d'utiliser les pointeurs dans le cadre de passage par adresse et de l'allocation dynamique (simple)

6 crédits ECTS

Volume horaire

Cours Magistral : 12h

Travaux Dirigés : 18h

Travaux Pratiques : 18h

Pré-requis nécessaires

Les cours d'informatique de 1ère année sont des pré-requis, notamment il est supposé que chaque étudiant-e :

- > possède des connaissances de base en algorithmique
- > a déjà programmé dans un langage impératif, idéalement proche du C (comme l'approche Java "impératif" du cours d'algorithmique et programmation de L1)
- > connaît les commandes de base d'un système Unix (se déplacer dans une arborescence, créer des dossiers/fichiers...)

Descriptif

Le contenu du cours est le suivant :

syntaxe élémentaire d'un programme C:

- > variables, tableaux, types primitifs
- > opérateurs de base
- > structures de contrôle : alternatives, boucles
- > fonctions

introduction aux types utilisateurs :

- > structure
- > de façon beaucoup plus brève : union, énumération

quelques rudiments de la bibliothèque standard C

- > fonctions d'entrée/sortie de base
- > fonctions sur les caractères et les chaînes de caractères
- > fonctions sur la manipulation de fichiers

introduction aux pointeurs :

- > déclaration des pointeurs, opérateurs * et &
- > passage par adresse des paramètres d'une fonction
- > arithmétique des pointeurs
- > pointeurs sur une structure (opérateur ->)
- > allocation dynamique (malloc, calloc, realloc, free)

Bibliographie

Le langage C - Norme ANSI - 2ème édition de Brian W. Kernighan et Dennis M. Ritchie (Éditions DUNOD).

Le guide complet du langage C de Claude Delannoy (Éditions Eyrolles).

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CC	Travaux Pratiques		1/3	
UE	CT	Ecrit - devoir surveillé	120	2/3	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	120	1/1	Pas de report de CC, même favorable.

Bloc transversal S3

6 crédits ECTS

Anglais

2 crédits ECTS

Volume horaire

Travaux Dirigés : 16h

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CC	Ecrit et/ou Oral		100/100	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	60	100/100	

Culture scientifique

2 crédits ECTS

Volume horaire

Cours Magistral : 6h

Travaux Dirigés : 10h

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CC	Ecrit et/ou Oral		50/100	
UE	CT	Ecrit - devoir surveillé	180	50/100	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	120	100/100	

Compétences numériques

2 crédits ECTS

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
Travaux Dirigés	Autre modalité	Pratique - examen en ligne		1	Validation conditionnée par passage de la certification PIX.

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
Autres	Autre modalité	Pratique - examen en ligne			Idem 1ère session.

Projet de conception objet

Présentation

Découverte du paradigme objet avec Python et une application projet

Découverte du paradigme objet (notion d'objet, d'héritage). Le langage Python sert de support à l'expérimentation de petits exemples et au développement d'une application plus conséquente à travers un projet. Le cours est divisé en deux parties

- > partie 1: concepts de base de l'objet et illustration au niveau des collections, de la récursivité, des graphes et du développement d'interface graphique
- > partie 2: mise en place d'un projet avec interface graphique permettant d'identifier les objets et leurs interactions.

6 crédits ECTS

Volume horaire

Cours Magistral : 18h

Travaux Dirigés : 18h

Travaux Pratiques : 18h

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CC	Travaux Pratiques		1/2	
	CT	Ecrit - devoir surveillé	120	1/2	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	120	1/1	

Réseau : modèle OSI

Présentation

- > Les concepts réseaux : historique, principaux éléments matériels et logiciels, technologies des réseaux, modèles de référence dont le modèle ISO
- > Les couches du modèle OSI et les protocoles associés.
- > Application pratique sur simulateurs de réseaux (GNS3 et/ou Cisco Packet Tracer et/ou Filius et/ou autre) des concepts présentés.

2 crédits ECTS

Volume horaire

Cours Magistral : 6h

Travaux Dirigés : 4h

Travaux Pratiques : 8h

Compétences visées

- > Connaître les concepts de bases des réseaux
- > Connaître le modèle OSI et son positionnement par rapport aux autres modèles de réseaux
- > Savoir configurer un réseau et les équipements réseaux

Bibliographie

- > Tanenbaum A, Wetherall D. Réseaux - 5ème édition. Pearson Education, 2011.
- > Pujolle G. Les réseaux - 5ème édition. E. Eyrolles, 2006

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
EC	CT	Ecrit - devoir surveillé	60	1/2	
EC	CC	Travaux Pratiques		1/2	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
EC	CT	Ecrit - devoir surveillé	60	1	pas de report de CC

Développement Web côté client

Présentation

Partie 1 : Conception de l'interface web côté client

- > Rappel HTML, CSS
- > Introduction au JavaScript

Partie 2 : Web framework : Bootstrap et jquery

Partie 3 : Web application côté client

- > Gestion de formulaires web
- > Vérification de données
- > Document object model (DOM)
- > Introduction au XML

3 crédits ECTS

Volume horaire

Cours Magistral : 6h

Travaux Dirigés : 4h

Travaux Pratiques : 10h

Pré-requis nécessaires

Programmer en HTML et en CSS (compétences acquises par la partie Web de l'UE Technologies Informatique, Portail ISI, Licence 1 ou de l'UE Introduction à l'informatique, Portail MPI, Licence 1) :

- Gestion de ressources (fichiers HTML, CSS, multimédia) d'un projet web
- Coder les balises HTML de base
- Coder les propriétés CSS de base
- Structurer les composants d'une page web selon sur une charte graphique précisée.

Compétences visées

Développer une meilleure compréhension de l'utilisation de HTML et CSS dans un contexte pratique et professionnel. Fournir plus d'interactions entre les utilisateurs et un site Web grâce à l'usage de JavaScript. Être capable de collecter et de vérifier les données des utilisateurs avec un formulaire Web. Avoir une première expérience de manipulation d'une base de données simple côté client.

- > Maîtriser l'utilisation de HTML/CSS et Javascript dans la conception de l'interface web côté client. Être capable d'utiliser et lire des documentation de frameworks CSS/JS comme jquery, Bootstrap.
- > Programmer en JavaScript
- > Construire des formulaires web et vérifier la validité des données fournies par l'utilisateur.
- > Gestion (ajouter, modifier, supprimer) de données dans une base de données sample côté client.

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	60	1/2	
UE	CC	Travaux Pratiques		1/2	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	120	1	pas de report de CC

Langages formels et automates

Présentation

Le cours présente les notions relatives aux langages réguliers et aux automates à états finis. On présente également des outils dans le cadre de l'analyse lexicale.

- > Introduction aux langages formels : alphabets, mots, facteurs, ordre sur les mots, opérations ensemblistes et algébriques sur les langages, fermetures des langages, langages réguliers, expressions régulières.
- > Introduction aux automates : automates déterministes et non déterministes, calculs dans un automate, langages reconnaissables, automates complets, produits d'automates, détermination des automates, automates asynchrones, théorème des éliminations des epsilon-transitions, exemples de langages reconnus, théorème de Kleene, l'algorithme de Brzozowski et Mac Cluskey, minimisation des automates. Exemples d'applications d'automates.

4 crédits ECTS

Volume horaire

Cours Magistral : 16h

Travaux Dirigés : 12h

Travaux Pratiques : 8h

Pré-requis nécessaires

notions d'algorithmique et de programmation étudiées en licence première année.

Compétences visées

- > Comprendre les notions générales relatives aux automates.
- > Être capable d'appliquer les méthodes de simplification ou de réduction associées automates.
- > Être capable d'établir les liens entre les automates et les langages réguliers.
- > Être capable de concevoir des analyseurs lexicaux.

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	90	1	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	90	1	

Algorithmique fondamentale, Graphes et Test

Présentation

- > Élaboration d'un schéma de résolution pour un problème donné (aspects données et contrôle).
- > Évaluation de la qualité d'un programme (complexité, test unitaire, preuve d'invariants).
- > Exploitation de la théorie des graphes, algorithmes d'optimisation.

6 crédits ECTS

Volume horaire

Travaux Pratiques : 16h

Cours Magistral : 20h

Travaux Dirigés : 20h

Pré-requis nécessaires

Connaissances de base en algorithmique et programmation impérative, en C et en Java, correspondant aux UE "Algorithmique et programmation" (S2) et "Programmation C" (S3).

Compétences visées

- > Traduire un énoncé de problème en spécification
- > Modéliser les données d'un problème
- > Ecrire un algorithme à partir d'une spécification
- > Tester unitairement un programme

Descriptif

Partie Algorithmique et test : acquisition des bases en algorithmique permettant d'élaborer un schéma de résolution pour un problème donné

- > Représentation des données (modèles formels de données, Types Abstraits de Données, implantation via structures de données associées), spécifications d'algorithmes adaptés.
- > Évaluation de la qualité des programmes produits (complexité, éléments de preuve de programme), test et validation d'une application simple (objectifs et méthodes de test, test unitaire, analyse et interprétation des résultats des tests, arrêt du test).

Partie Graphes :

- > Introduction aux notions de Relations et Graphes (applications, terminologie et représentation).
- > Résolution de problèmes de cheminement et connexité (chemins eulériens, hamiltoniens, postier chinois, composantes connexes), algorithmes de plus court chemin (Dijkstra, Floyd, Ford-Bellman, A*),
- > modélisation et résolution de problèmes classiques d'optimisation (recouvrement, flots, coloration, partitionnement ordonnancement).

Bibliographie

- > *Concepts fondamentaux de l'informatique*, A. AHO, J. ULLMAN, ed: DUNOD
- > *Optimisation combinatoire T2: programmation discrète*, M. Sakarovitch, Hermann, 1984
- > *Méthodes d'optimisation*, I. Charon et al., Masson, 1996
- > *Introduction à l'algorithmique*, T.H. CORMEN, C.E. LEISERSON & R.L. RIVEST, Dunod
- > *Le test des logiciels*, S. Xanthakis, P. Régnier et C. Karapoulios, Hermès, 2000.

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CC	Travaux Pratiques		1/3	
UE	CT	Ecrit - devoir surveillé	120	2/3	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	120	1	Pas de report de CC, même favorable.



Université de Bretagne Occidentale

Langue d'enseignement

Français

Architecture et Systèmes 1

Présentation

Partie 1 : Conception

La première partie du cours mène à la conception d'un petit processeur très basique afin d'en comprendre les briques de base dans une architecture très élémentaire. En partant des portes logiques de base et de la représentation de l'information au plus bas niveau, nous voyons la conception d'unités de calcul et de mémorisation élémentaires, puis d'une unité de contrôle (câblée ou microprogrammée) modélisée à l'aide d'un automate, menant à la réalisation du petit processeur.

Partie 2 : Programmation Assembleur

Cette partie est une très brève introduction à la programmation en langage d'assemblage d'un processeur industriel, afin de voir un aperçu de la programmation au plus bas niveau d'une mise en œuvre réelle de la partie conception. De plus, cette partie permet d'expliquer certains mécanismes comme le déroulement complet d'un appel de fonction ou comment se traduisent les instructions de contrôles (branchements, boucles) au plus bas niveau.

Partie 3 : Programmation système

Dans cette dernière partie on revient sur certains éléments de base d'un système d'exploitation. La notion de processus est présentée, avec une présentation de la génération des processus et une première introduction au parallélisme via des processus (lourds) parallèles, et des premiers outils de coopération inter processus avec l'aide de sémaphores.

6 crédits ECTS

Volume horaire

Travaux Pratiques : 18h

Cours Magistral : 12h

Travaux Dirigés : 18h

Pré-requis nécessaires

Partie Unix (Environnements informatiques) 1ère année, Programmation C (S3), portes logiques et rudiments de logique booléenne (1ère année)

Compétences visées

Ce cours a pour objectif d'améliorer la compréhension des couches basses (couches matérielle et système d'exploitation) via :

- > une compréhension élémentaire des éléments constitutifs d'un microprocesseur séquentiel simple
- > une brève introduction à un langage d'assemblage (en particulier pour détailler le fonctionnement d'un appel de fonction) et illustrer le contrôle d'un processeur au plus bas niveau de programmation
- > une introduction aux systèmes d'exploitation avec un focus sur les processus (norme POSIX) : génération de processus, caractéristiques de base, coopération inter processus (sémaphores)

Bibliographie

- > Architecture des machines et des systèmes informatiques, Joëlle Delacroix et Alain Cazes, Dunod
- > Développement système sous Linux, Christophe Blaess, Editions Eyrolles
- > Architectures logicielles et matérielles, Paul Amblard & co, Dunod
- > Organisation et architecture de l'ordinateur, William Stallings, Pearson Education France

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	120	2/3	
UE	CC	Travaux Pratiques		1/3	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	120	1	Pas de report de CC, même favorable.

Logique

Présentation

Ce cours présente les éléments de la logique booléenne, le calcul propositionnel, le calcul des prédicats, les ensembles et les relations. On présente également des outils dans le cadre de la vérification formelle de programmes.

- > Systèmes formels et logiques. Syntaxe et logique des propositions, sémantique, procédures de démonstration (fonction de vérités, méthodes axiomatiques, déduction naturelle, séquents de Gentzen, résolution).
- > Ensembles : construction par énumération et induction, appartenance, union, intersection, produits cartésiens, relations. Récursion, principes d'induction. Ensembles et fonctions définis inductivement. Applications : introduction à la théorie des langages, notion de typage.

2 crédits ECTS

Volume horaire

Cours Magistral : 8h

Travaux Dirigés : 12h

Pré-requis nécessaires

notions de mathématiques étudiées en licence première année.

Compétences visées

- > Comprendre les concepts fondamentaux de la logique.
- > Etre capable de formaliser des problèmes dans le cadre du calcul des prédicats.

Bibliographie

1. *Introduction à la logique*. R. David, K. Nour, C. Raffali. Editions Dunod.
2. *Concepts fondamentaux de l'informatique*. A Aho, J. Ullman. Dunod

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	90	1	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	90	1	

Programmation fonctionnelle

Présentation

Approche fonctionnelle de la programmation, récursivité. Présentation d'un langage fonctionnel.

6 crédits ECTS

Volume horaire

Travaux Dirigés : 18h

Cours Magistral : 18h

Travaux Pratiques : 18h

Descriptif

- > Types, fonctions, récursivité. Terminaison, validité et complexité des programmes récursifs.
- > Structures de données: listes, piles, files, arbres, graphes.
- > Algorithmes classiques et avancés sur ces structures de données.
- > Évaluations par valeur et paresseuse. Programmation modulaire.
- > Présentation d'un langage fonctionnel (OCaml).

Bibliographie

Le langage Caml, X. Leroy et P. Weiss. Masson

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CC	Travaux Pratiques	90	1/3	
UE	CT	Écrit - devoir surveillé	120	2/3	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Écrit - devoir surveillé	120	1	Pas de report de CC, même favorable.

Java 1

Présentation

Ce cours reprend trois concepts fondamentaux de la programmation objet abordés en Python (l'encapsulation, la composition et l'héritage) et les développe dans le contexte technique différent d'un langage de programmation impératif et typé statiquement : le langage Java. L'objectif est triple :

- > mettre en évidence l'intérêt essentiel de la programmation objet, i.e. sa nature intrinsèquement modulaire et extensible
- > donner les premiers éléments méthodologiques de conception qui permettent de déterminer les objets utiles à la réalisation d'un programme en précisant leurs responsabilités et leurs interactions
- > consolider la maîtrise du langage Java, de ses constructions élémentaires, et de son API standard

L'UE est organisée en 3 leçons comportant chacune un cours, une séance de travaux dirigés et une à deux séances de travaux pratiques :

1 Encapsulation : modèle de données structurées, enfouissement, initialisation, consultation, mise à jour, interface d'interactions

2 Composition : références d'objet, changement d'état, interaction, délégation

3 Héritage : extension de structure, extension de comportement, mise à jour de comportement, liaison dynamique, polymorphisme

2 crédits ECTS

Volume horaire

Travaux Dirigés : 6h

Travaux Pratiques : 8h

Cours Magistral : 6h

Pré-requis nécessaires

Pour aborder ce cours, il est utile d'avoir déjà programmé dans un langage de programmation procédural et d'avoir un minimum de connaissances sur :

- > les types, les variables, les opérateurs
- > les instructions de contrôle (appels, boucles, conditionnelles)
- > l'algorithmique élémentaire (tri, recherche de max, calcul, etc.)

L'UE Algorithmique et Programmation de L1 fournit ces pré-requis.

Compétences visées

Traduire la spécification d'un programme en système d'objets en assignant des responsabilités.

Concevoir un programme à la fois ouvert à l'extension et fermé à la modification.

Mobiliser à bon escient les constructions du langage Java et de son API.

Bibliographie

- Mickaël Kerboeuf : Fondements de la programmation orientée objet avec Java 8. Références Sciences, Ellipses, décembre 2016, ISBN 9782340014824.

- Mickaël Kerboeuf : Algorithmique et programmation objet. Références Sciences, Ellipses, mai 2020, ISBN 9782340037984.

- James Gosling, Bill Joy, Guy Steele et Gilad Bracha : The Java(TM) Language Specification (3rd Edition). Addison-Wesley Professional, 2005, ISBN 0321246780.

- Erich Gamma, Richard Helm, Ralph Johnson et John Vlissides : Design Patterns, Elements of Reusable Object-oriented Software. Addison-Wesley Longman Publishing Co., 1995, ISBN 0-201-63361-2.

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	90	2/3	
UE	CC	Travaux Pratiques		1/3	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	90	1	

Developpement Web côté serveur

Présentation

- > Rappels : BD et SGBD.
- > Respect d'un cahier des charges.
- > Introduction à l'architecture client/serveur.
- > Introduction à la conception d'une base de données relationnelle (diagramme UML de classes) puis passage du modèle conceptuel au modèle logique (schéma relationnel).
- > Création des tables dans la base de données relationnelle, insertion du jeu de données et manipulation des données (langage SQL).
- > Interfaçage et manipulation des données d'une base de données relationnelle (développement Web côté serveur : création de formulaires et de pages Web).

2 crédits ECTS

Volume horaire

Cours Magistral : 8h

Travaux Dirigés : 8h

Travaux Pratiques : 10h

Pré-requis nécessaires

- > Création de pages Web statiques (HTML / CSS / Javascript)
- > Création d'une base de données relationnelle, insertion du jeu de données et manipulation des données

Compétences visées

- > Concevoir et implanter une base de données sur un SGBD relationnel (mysql, Oracle ...) en utilisant le langage SQL.
- > Manipuler les données de la base de données relationnelle en utilisant le langage SQL.
- > Développer, côté serveur, l'application Web permettant de manipuler les données de la base de données.
- > Travailler en mode projet.

Bibliographie

- > J.P. Hainaut "Bases de données" (Dunod)
- > Brouard "SQL" (CampusPress)

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	60	2/3	
UE	CC	Travaux Pratiques		1/3	report en session 2 si favorable

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	60	1	report de CC (coef 1/3) si favorable à l'étudiant

Langue d'enseignement

Français

Bloc transversal S4 I

6 crédits ECTS

Anglais

2 crédits ECTS

Volume horaire

Travaux Dirigés : 16h

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	60	80/100	
UE	CC	Ecrit et/ou Oral		20/100	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	60	100/100	

Option transversale S4

2 crédits ECTS

Volume horaire

Travaux Dirigés : 11h

Culture scientifique

2 crédits ECTS

Volume horaire

Travaux Dirigés : 16h

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CC	Oral	15	100%	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Oral	15	100%	

Sea-EU / Sport, culture, engagement étudiant

2 crédits ECTS

Préparation à l'Expérience Professionnelle en Synergie en Informatique

Présentation

L'UE, comptant pour le S4, comporte 6h de TD au S3 et 10h au S4

Au S3, Interview d'un professionnel exerçant dans la région brestoise : 6h TD destinées à cibler un professionnel dont l'activité correspond aux compétences acquises à l'issue d'un cursus en informatique. Recherches menées en cours à partir de LinkedIn et des annuaires d'anciens étudiants diplômés de masters et insérés récemment

professionnellement. Préparation des questions en vue des interviews.

Au S4, 6h TD destinées à comprendre les enjeux du travail en équipe à travers des notions de management et l'élaboration de feuilles de route afin de faciliter l'organisation des tâches. Puis 4h de présentations orales évaluées en contrôle continu par groupe à partir des interviews menées à l'issue des cours du S3.

2 crédits ECTS

Volume horaire

Travaux Dirigés : 16h

Objectifs

> S3: acquérir des connaissances sur le monde professionnel dans le secteur de l'informatique.

> S4: savoir travailler et collaborer en équipe. Etre capable de mener à son terme un projet en informatique en répondant à la demande du client.

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CC	Autre nature		1	CC à définir

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	Report de notes	Autre nature		1	report de session 1

Programmation C avancée

Présentation

Le but de cette UE est d'approfondir les connaissances des étudiants en C et d'appliquer ces connaissances dans le cadre d'un projet.

6 crédits ECTS

Volume horaire

Travaux Pratiques : 32h

Travaux Dirigés : 12h

Cours Magistral : 10h

Objectifs

- > Maîtriser la syntaxe et de la sémantique du langage C.
- > Savoir développer et déboguer une application en langage C.

Pré-requis nécessaires

Connaissances de l'algorithmique et de la programmation en C, correspondant aux UE [Algorithmique et programmation](#) et [Langages de programmation](#) (parcours IFA), ou à l'UE [C, Algorithmique et Structures de Données](#) (parcours CDA).

Compétences visées

L' étudiant doit disposer, parmi l'ensemble de ceux qu'il maîtrise, d'un langage adapté à la programmation de systèmes disposant d'un support d'exécution restreint en terme des ressources matérielles et/ou de services logiciels disponibles.

Descriptif

1. Langage C avancé :

Rappel sur les classes d'allocations.

Codage de structures de données complexes.

Pointeurs de fonctions, fonctions "génériques".

Conversions de types, implicites et explicites, portabilité du code.

Fonctions à nombre variables d'arguments, mécanisme de passage. des arguments à une fonction.

E/S haut niveau, contrôle des périphériques.

2. Environnement de programmation :

Compilation séparée.

Outils make, déboggeur symbolique.

Éditions de liens, construction et utilisation de bibliothèques statiques et dynamiques.

Introduction à l'internationalisation.

3. Développement d'une application simple en langage C, intégrant si besoin est, une IHM graphique élémentaire (GTK) et un accès à une base de données.

Bibliographie

1. B. Kernighan, R. Pike, **La programmation en pratique**, Vuibert informatique, 2001
2. A. Braquelaire, **Méthodologie de la programmation en C (Norme C 99 - API POSIX)**, Dunod, 2005
3. B.W. Kernighan, R. Pike, David Odin, **Programmation Linux avec GTK+**, 2000, Eyrolles

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CC	Travaux Pratiques		1/2	
UE	CT	Ecrit - devoir surveillé	120	1/2	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	120	1	Pas de report de CC, même favorable.

Programmation logique et Intelligence artificielle

Présentation

Ce cours comprend une introduction à la programmation logique et à la programmation par contraintes, ainsi qu'une initiation à quelques aspects de l'intelligence artificielle. Il comprend 4 parties :

- > Programmation logique (6h CM, 6h TD, 4h TP) : fondements, représentations de données, principes algorithmiques.
- > Programmation par contraintes (4h CM, 4h TD, 4h TP) : modélisation et résolution de problèmes.
- > Résolution de problèmes avec l'algorithme A*, heuristiques (4 h CM, 4h TD, 4h TP)
- > Introduction théorique sur les jeux de stratégie impartiaux (jeux de Nim, théorème de Sprague-Grundy). Développement d'un joueur pour un jeu de stratégie à deux joueurs : algorithme minimax, élagage alpha-bêta. (4h CM, 4h TD, 6h TP)

6 crédits ECTS

Volume horaire

Travaux Dirigés : 18h

Travaux Pratiques : 18h

Cours Magistral : 18h

Objectifs

Partie PLC:

- > Connaître les bases d'un langage de programmation logique.
- > Savoir programmer des algorithmes de base (avec listes, arithmétique, coupure) en programmation logique.
- > Connaître les bases de la programmation par contrainte, et savoir résoudre un problème à l'aide de la programmation par contrainte.

Partie IA:

- > Connaître certains aspects de l'intelligence artificielle: exploration (algorithme minimax, A*...), joueurs automatiques.

Pré-requis nécessaires

Connaissances en algorithmique et programmation, initiation à la programmation fonctionnelle.

Compétences visées

Ecrire ou dérouler un programme simple en programmation logique. Modéliser puis résoudre un problème en programmation par contraintes. Résoudre des problèmes avec des l'algorithme A*. Etre familiarisé avec des aspects de la théorie des jeux, l'algorithme minimax et ses améliorations.

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CC	Travaux Pratiques		1/3	
UE	CT	Ecrit - devoir surveillé	120	2/3	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	120	1	Pas de report de note de CC, même favorable.

Langue d'enseignement

Français

Ingénierie des systèmes d'information

Objectifs

- > Conception et la réalisation d'une application Web utilisant une base de données avec un framework MVC,
- > Modélisation de la base avec UML,
- > Mise en œuvre d'une base de données relationnelle normalisée,
- > Interrogation des bases de données (SQL, Fonction, Trigger, Curseur,...),
- > Gestion des transactions.
- > Préparation d'un plan de test de validation

6 crédits ECTS

Volume horaire

Travaux Pratiques : 18h

Travaux Dirigés : 18h

Cours Magistral : 18h

Descriptif

- > Mise en œuvre d'une base de données relationnelle normalisée,
 - > 3 premières formes normales
 - > algèbre relationnelle
- > Utilisation du diagramme de classes UML pour concevoir une base de données,
- > Langage SQL :
 - > Langage de Définition de Données (LDD)
 - > Langage de Manipulation de Données (LMD)
 - > SQL avancé : vue, procédure, trigger,
- > Gestion des transactions avec SQL
- > Programmation d'une application Web MVC en utilisant les fonctions d'une bibliothèque PHP et un framework PHP
- > Tests unitaires
- > Réalisation des tests de validation.

Bibliographie

- > Jean-Luc HAINAUT, Bases de données, Dunod 2009
- > F. BROUARD, R. BRUCHEZ, C. SOUTOU, "SQL", Pearson, 2012
- > Christian SOUTOU, Modélisation des Bases de Données, Eyrolles, 2015
- > Pascal ROQUES, UML2 par la pratique, Eyrolles, 2005

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CC	Travaux Pratiques		40%	
UE	CT	Ecrit - devoir surveillé	120	60%	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	120	100%	Pas de report de CC (même favorable).

Java et Conception d'Applications

Présentation

UE constituée de deux EC obligatoires : Java 2 et Conception d'Applications.

6 crédits ECTS

Java 2

Présentation

Ce cours revient sur les concepts fondamentaux de la programmation objet : l'encapsulation, la composition et l'héritage. En complément, ce cours aborde les concepts de classe enfouie, de classe abstraite, d'interface, d'introspection, de clonage, d'exception, de généricité et de lambda-expression. Le langage Java sert de support à l'apprentissage de ces concepts. Le cours aborde également les éléments indispensables de l'API pour le développement d'applications en Java comme les collections, les flots de données ou les fils d'exécution. Enfin, le cours présente les principaux outils de développement associés à Java : outils de tests unitaires et de non-régression, générateurs de documentation, outils de contrôle de la qualité du code, et outils de travail collaboratif.

L'UE est organisée en 10 leçons comportant chacune un cours, une à deux séances de travaux dirigés et une à deux séances de travaux pratiques :

1. Rappels de Java : fondements impératifs, fondements Objet (encapsulation, composition, héritage)
2. Classes enfouies, classes abstraites, interfaces
3. Introspection, clonage, égalité, représentation littérale
4. Exceptions : principes généraux et mise en oeuvre en Java
5. Généricité : polymorphisme et types génériques
6. Objets fonctionnels : lambda-expressions
7. Collections : API standard, collections simples et tableaux associatifs
8. Flots de données : flots d'octets et de caractères, gestion de fichiers et analyse lexicale
9. Fils d'exécution : principes élémentaires de programmation concurrente
10. Outils pour le développement d'applications : JUnit, Javadoc, Doxygen, Checkstyle, SVN, Git

Objectifs

L'objectif est quadruple :

- > savoir tirer parti de la programmation objet pour la conception et la maintenance de programmes modulaires, réutilisables, évolutifs et sûrs
- > compléter les éléments méthodologiques d'analyse, de conception et de programmation orientées objet
- > renforcer la maîtrise du langage Java, de ses constructions élémentaires, et de son API standard
- > découvrir l'outillage essentiel au développement d'une application d'envergure

- > maîtriser le langage Java : sa syntaxe et ses fondements sémantiques
- > connaître les différentes phases de la compilation d'un programme Java
- > connaître le mécanisme d'exécution d'un programme Java et le fonctionnement de la machine virtuelle
- > maîtriser les rudiments de la programmation objet
- > connaître quelques bonnes pratiques de conception objet

D'autre part, concernant l'outillage pour la conduite des projets, les étudiants sont amenés à expérimenter et à comprendre l'intérêt et les principes des outils :

- > de validation par tests unitaires
- > de documentation logicielle
- > de contrôle de la qualité du code
- > de gestion de versions et travail collaboratif

Pré-requis nécessaires

Pour aborder ce cours, il est nécessaire d'avoir déjà programmé en Java ou dans un langage de programmation Objet similaire, et d'avoir un minimum de connaissances sur :

- > les aspects procéduraux de Java (types, variables, opérateurs, appels, boucles, conditionnelles) et l'algorithmique élémentaire (tri, recherche de max, calcul, etc.)
- > les fondements de la programmation Objet (encapsulation, composition, héritage)

4 crédits ECTS

Volume horaire

Travaux Pratiques : 12h

Cours Magistral : 12h

Travaux Dirigés : 12h

- > les principes élémentaires de conception Objet (identification des classes, assignation de responsabilités, structure, composition et comportement des objets)

L'UE Java 1 de L2 et l'UE Introduction à l'objet et aux systèmes d'exploitation et microprocesseur en L3 fournissent ces pré-requis.

Compétences visées

Traduire la spécification d'un programme en système d'objets en assignant des responsabilités.

Concevoir un programme à la fois ouvert à l'extension et fermé à la modification.

Mobiliser à bon escient les constructions du langage Java et de son API.

Savoir utiliser les outils de tests unitaires, les générateurs de documentation, les outils de contrôle de la qualité du code, et les outils de travail collaboratif.

Bibliographie

- > Mickaël Kerboeuf : Fondements de la programmation orientée objet avec Java 8. Références Sciences, Ellipses, décembre 2016, ISBN 9782340014824.
- > Mickaël Kerboeuf : Algorithmique et programmation objet. Références Sciences, Ellipses, mai 2020, ISBN 9782340037984.
- > James Gosling, Bill Joy, Guy Steele et Gilad Bracha : The Java(TM) Language Specification (3rd Edition). Addison-Wesley Professional, 2005, ISBN 0321246780.
- > Erich Gamma, Richard Helm, Ralph Johnson et John Vlissides : Design Patterns, Elements of Reusable Object-oriented Software. Addison-Wesley Longman Publishing Co., 1995, ISBN 0-201-63361-2.

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	120	2/3	
UE	CC	Travaux Pratiques		1/3	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	120	1	Pas de report de CC, même favorable.

Conception d'applications

Présentation

Cette UE a pour objectif la maîtrise des outils du développeur (codage, test et déploiement) pour réaliser une application déployable et utilisable par un utilisateur final. L'application visée couvre plusieurs technologies (développement objet, développement Web, base de données).

L'UE est organisée sous forme de projet à réaliser en groupe de 4 à 5 étudiants selon une méthode itérative basée sur plusieurs Sprints.

2 crédits ECTS

Volume horaire

Cours Magistral : 2h

Travaux Dirigés : 6h

Travaux Pratiques : 12h

Pré-requis nécessaires

Programmation Java. Programmation Web. Programmation Base de données. Mise en place de tests unitaires.

Compétences visées

Savoir développer une application. Savoir spécifier et mettre en place un IHM. Savoir décrire et mettre en place une base de données. Savoir décrire et documenter le code. Maîtriser la définition et la mise en place de tests unitaires. Maîtriser la qualité du code. Maîtriser les outils du développeur (IDE, suivi de version, vérification de la qualité du code, spécification du code, mise en œuvre de tests unitaires).

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
EC	CC	Travaux Pratiques		1	note reportée en session 2

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
EC	Report de notes	Travaux Pratiques		1	report de session 1

Bloc transversal S5 - IFA**6 crédits ECTS**

Anglais S5

2 crédits ECTS

Volume horaire

Travaux Dirigés : 16h

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
Travaux Dirigés	CC	Ecrit et/ou Oral		30/100	
Travaux Dirigés	CT	Ecrit - devoir surveillé	90	70/100	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
Travaux Dirigés	CT	Ecrit - devoir surveillé	60	100/100	

Communication S5

2 crédits ECTS

Volume horaire

Travaux Dirigés : 12h

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CC	Ecrit et/ou Oral		100/100	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	60	100/100	

Option ouverture Master (1 au choix)

Présentation

Option d'ouverture Master.

2 crédits ECTS

Administration système

2 crédits ECTS

Volume horaire

Cours Magistral : 8h

Travaux Dirigés : 6h

Travaux Pratiques : 8h

Descriptif

- > Rôle d'un administrateur système
- > Description d'un réseau Sécurité informatique
- > Tâches d'administration d'un système Unix
- > Cryptographie
- > Automatisation des tâches d'administration

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CC	Travaux Pratiques		1/3	
UE	CT	Travaux Pratiques	60	2/3	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Oral	15	1	Aucun report de note.

Objets connectés et robotique

Présentation

Descriptif

- > Ce module propose une introduction aux objets connectés au travers d'expérimentations mettant en jeu des robots mobiles et des objets connectés et communicants. Le projet des étudiants sera orienté vers des applications concrètes, sur des petits robots mobiles programmables.
- > La plate-forme expérimentale permet d'aborder les aspects mécaniques (châssis, moteurs, ...), les capteurs et les actionneurs (accéléromètres, magnétomètres, mesure de distances, ...), les langages pour la gestion des capteurs et des actionneurs (C), les processeurs (Arduino) et la communication sans fil (XBee).
- > Les missions que le robot doit effectuer sont contrôlées à distance depuis un PC (avancer jusqu'à un obstacle, contourner un obstacle, longer un mur, ...). Un protocole de communication sera défini pour les échanges entre le PC et le robot (envoi des ordres au robot, transmission des mesures des capteurs du robot au PC). Le JSON (Javascript Object Notation) sera utilisé pour les échanges PC-robot.
- > Sur le PC, on utilisera le langage NodeJS et on abordera les notions de gestion des événements (messages en provenance du robot) et de programmation modulaire. On aura un module pour gérer l'IHM Web (boutons pour commander le robot, affichage des mesures des capteurs), un module pour gérer le robot et un module pour gérer la liaison sans fil (XBee). Pour gérer les différentes missions et les enchaîner, on aura en plus des modules spécialisés

2 crédits ECTS

Volume horaire

Cours Magistral : 8h

Travaux Pratiques : 14h

Compétences visées

Introduction à la thématique des objets communicants, parmi lesquels des robots

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CC	Travaux Pratiques		1	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Oral	20	1	Aucun report de note.

Sécurité

Présentation

L'objectif est de former aux bases de la sécurité des systèmes informatiques et des systèmes dont le fonctionnement repose sur l'utilisation des technologies de l'information.

- > Notions de base de la sécurité informatique, menaces, techniques d'attaques, mécanismes de protection
- > Simulation d'attaques
- > Outils de détection d'attaques
- > Mise en œuvre pratique de mécanismes de sécurité.

2 crédits ECTS

Volume horaire

Cours Magistral : 8h

Travaux Dirigés : 6h

Travaux Pratiques : 8h

Compétences visées

- > Connaître les concepts de base de la sécurité informatique
- > Connaître les menaces auxquelles sont exposés les systèmes informatiques et les systèmes dont le fonctionnement repose sur l'utilisation des technologies de l'information
- > Connaître les principales approches permettant de prévenir et de détecter les attaques contre les systèmes
- > Savoir mettre en œuvre des solutions de sécurité de systèmes informatiques

Bibliographie

- > ACISSI. Sécurité informatique – Ethical Hacking : apprendre l'attaque pour mieux se défendre, 5ème édition. Editions ENI, 2017
- > Stallings W. Network security essentials, applications and standards, Third Edition. Pearson Prentice Hall, 2007

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CC	Travaux Pratiques		1	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Oral	15	1	

Projet applicatif

Présentation

Cette UE consiste en un ou deux projets développés en intensif, par groupe de 2 à 4, pendant deux semaines en continu. L'objectif pédagogique est d'apprendre à élaborer un développement applicatif adéquat d'un système et de savoir le présenter. L'objectif pratique du problème à résoudre est ambitieux. Il s'agit d'un projet transversal, tourné vers la maturation et le croisement des connaissances acquises pendant la licence. Le problème peut potentiellement être résolu de façon diverse pour valoriser la créativité. La démarche pédagogique consiste à placer les étudiants dans une situation de recherche, dans laquelle ils doivent s'investir, se poser des questions, faire des choix, trouver des solutions et apprendre à les argumenter.

Le cahier des charges est présenté en début de semaine par un énoncé général complété par un ensemble de sous-problèmes à résoudre et la présentation des artefacts à produire. Cette présentation peut éventuellement introduire une chronologie à respecter pour la résolution des sous-problèmes, des exemples et des démonstrations fonctionnelles.

En fin de projet, les étudiants doivent être capables de présenter leur mise en œuvre au cours d'un exposé de 15 à 20 minutes avec démonstration.

3 crédits ECTS

Volume horaire

Travaux Pratiques : 48h

Autres : 22h

Objectifs

- > Travailler en équipe et en réseau ainsi qu'en autonomie et responsabilité au service d'un projet;
- > Choisir, sur des critères objectifs, les structures de données et construire les algorithmes les mieux adaptés à un problème donné;
- > Analyser et interpréter les résultats produits par l'exécution d'un programme;
- > Appliquer des approches raisonnées de résolution de problèmes complexes par décompositions et/ou approximations successives et mettre en œuvre des méthodes d'analyse pour concevoir des applications et algorithmes à partir d'un cahier des charges partiellement donné;
- > Concevoir le traitement informatisé d'informations de différentes natures, telles que des données, des images et des textes;
- > Expliquer et documenter la mise en œuvre d'une solution technique.

Pré-requis nécessaires

Pour aborder cette UE, il faut avoir suivi toutes les UE de la licence. En particulier celles qui visent directement le développement d'applications :

- > L'algorithmique et les structures de données;
- > Les langages de programmation comme le C, Python et Java;
- > L'ingénierie des systèmes d'information et la conception d'application;
- > Le développement des IHM.

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CC	Travaux Pratiques		1	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	Report de notes	Autre nature		1	

Algorithmique avancée

Présentation

partie optimisation (10h CM, 12h TD, 10h TP) :

Ce cours présente des approches par programmation linéaire, séparation & évaluation, programmation dynamique et heuristiques pour la résolution de problèmes d'optimisation combinatoire. Une introduction à la complexité et à la résolution de problèmes de satisfiabilité booléenne est également présentée.

Le cours aborde les points suivants :

- > complexité algorithmique et théorie de la complexité
- > modélisation et heuristique de résolution de problème de satisfiabilité booléenne (SAT)
- > programmation linéaire : méthode du simplexe, modélisation MILP de problèmes d'optimisation
- > méthodes par séparation et évaluation, application à la programmation linéaire, problème de voyageur de commerce
- > programmation dynamique : principe et applications (plus court chemin, gestion de stock)
- > Approches diviser pour régner (et calcul parallèle)
- > méthodes heuristiques approchées : introduction aux meta heuristiques, heuristiques gloutonnes. Applications d'heuristiques gloutonnes (sac à dos, couverture, voyageur de commerce)
- > méthodologie de comparaison d'algorithmes

La mise en pratique se fait avec le logiciel CPLEX :

- > utilisation directe
- > langage de programmation OPL et modélisation
- > génération de cas et automatisation de séries de test

partie Modélisation et programmation des réseaux sans fil (6h CM, 4h TD, 6h TP) :

Ce cours permet de présenter les notions de base de la modélisation et la programmation distribuée appliquée aux graphes et aux réseaux sans fil. Les simulations sont réalisées sur l'outil CupCarbon en utilisant le script SenScript.

Le cours présente :

1. Introduction au concept de la programmation distribuée et la présentation des algorithmes de base tels que le Local Minimum Finding (LMF) et le Global Minimum Finding (MinFinding) et les algorithmes de BFS/DFS distribués ainsi que le FLF (Flooding for Leaf Finding), pour trouver les feuilles d'un arbre couvrant.

Ce cours permet aussi de faire la distinction entre la programmation classique, la programmation parallèle et la programmation distribuée.

2. Présentation de la plateforme CupCarbon et du script SenScripr pour la programmation des noeuds distribués ou capteurs
3. Présentation de plusieurs variantes de l'algorithme de l'élection du leader dans un graphe euclidien connecté ou dans un réseau sans fil : LOGO, BrOGO, DoTRO et WBS
4. Présentation des Pseudo-Polygones
5. Présentation d'algorithmes distribués pour trouver les noeuds bordures internes et externes d'un graphe euclidien connecté. Ce cours introduit en premier les algorithmes classiques pour trouver l'enveloppe convexe dans un graphe non connecté (Algorithme de Jarvis) et l'enveloppe polygonale dans un graphe connecté (LPCN). Ensuite, il présente 2 versions distribuées du LPCN : D-LPCN et D-RRLLPCN

5 crédits ECTS

Volume horaire

Cours Magistral : 16h

Travaux Pratiques : 16h

Travaux Dirigés : 16h

Objectifs

L'objectif de cette unité d'enseignement est de donner une introduction à l'algorithmique avancée avec des applications dans le domaine des réseaux sans fil

Pré-requis nécessaires

théorie des graphes et logique propositionnelle

Bibliographie

Méthodes d'optimisation combinatoire, I.Charon, A.Germa, O.Hudry
Optimisation combinatoire (tome programmation discrète), M.Sakarovitch

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
Travaux Pratiques	CC	Travaux Pratiques		1	pour la partie modélisation et programmation des réseaux sans fil
UE	CC	Travaux Pratiques		1	pour la partie optimisation
UE	CT	Ecrit - devoir surveillé	120	2	pour la partie optimisation

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	120	1/1	Pas de report de CC de session 1. Pour la partie réseau sans fil et la partie optimisation (1h15 optimisation, 45mn réseau sans fil)

Architectures et Systèmes 2

6 crédits ECTS

Architectures 2

Présentation

Le cours décrit deux structures fondamentales utilisées dans les processeurs actuels : le pipeline et la hiérarchie mémoire. La problématique de la mesure des performances d'un processeur est introduite.

- > technique de pipeline appliquée à la réalisation d'un processeur RISC de première génération : principe, aléas de données, de contrôle, structurelle, prédiction de branchement
- > hiérarchie mémoire : cache et mémoire virtuelle
- > mesures de performances

Au cours des travaux pratiques, les étudiants sont invités à modéliser une partie d'un processeur RISC *pipeliné* en utilisant le langage de description de matériel VHDL, puis à simuler le comportement de ces modèles.

3 crédits ECTS

Volume horaire

Travaux Pratiques : 10h

Travaux Dirigés : 8h

Cours Magistral : 10h

Compétences visées

- > Comprendre la signification des caractéristiques techniques et de performances d'un ordinateur
- > Evaluer l'adéquation d'un algorithme et une architecture matérielle de processeur

Bibliographie

1. D. Patterson, J. Hennessy, *Organisation et conception des ordinateurs (l'interface matériel/logiciel)*, Dunod
2. *Operating Systems : Internals and Design Principles*, William Stallings - Prentice Hall
3. *Programmation systèmes en C sous Linux: Signaux, Processus, threads, IPC et sockets*, Christophe Blaess, Eyrolles
4. *Unix, programmation et communication*, Jean-Marie Rifflet, Dunod

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CC	Travaux Pratiques		1/3	
UE	CT	Ecrit - devoir surveillé	90	2/3	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	60	1	Pas de report de CC (même favorable).

Systèmes 2

Présentation

- > programmation *multi-threads*
- > ordonnancement,
- > systèmes de fichiers et gestion des ressources de mémoire
- > utilisation en langage C des primitives systèmes

3 crédits ECTS

Volume horaire

Cours Magistral : 10h

Travaux Dirigés : 10h

Travaux Pratiques : 8h

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
EC	CC	Travaux Pratiques		1/3	
EC	CT	Ecrit - devoir surveillé	90	2/3	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
EC	CT	Ecrit - devoir surveillé	60	1	Pas de report de CC (même favorable)

Réseaux IP

8 crédits ECTS

Réseaux IP : programmation des réseaux

Présentation

Programmation réseau en C

- > architecture des réseaux informatiques, adressage des machines
- > modèles réseau en couches : principes, couches et protocoles réseau,
- > protocoles TCP et UDP, notion de socket
- > programmation en C d'applications clientes et serveur à l'aide des primitives système et réseau (exemple de champ d'application possible : échange de données avec des objets connectés).

3 crédits ECTS

Volume horaire

Travaux Pratiques : 9h

Travaux Dirigés : 9h

Cours Magistral : 10h

Pré-requis nécessaires

- > Architecture et système (S4)
- > Programmation C avancée (S5)

Compétences visées

- > connaître le modèle client-serveur utilisé dans le cadre des SE distribués,
- > connaître les caractéristiques des protocoles TCP et UDP et être capable de programmer des applications de type client/serveur,
- > être sensibilisé à l'architecture d'un réseau local et à l'interconnexion des réseaux (ex : Internet).

Bibliographie

1. "Unix, programmation et communication", Jean-Marie Rifflet, Dunod
2. "TCP/IP : architecture, protocoles, applications" - D. Comer - Dunod
3. "Cours Réseaux et télécoms" - G. Pujolle - Eyrolles
4. "Cours Réseaux" - V. Ribaud et P. Le Parc - Polycoopié de cours du département Informatique, UBO, Brest.

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CC	Travaux Pratiques		1/3	
UE	CT	Ecrit - devoir surveillé	60	2/3	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	60	1	Pas de report de CC (même favorable).

Réseaux IP : RPC

Présentation

Introduction aux systèmes répartis

- > Définitions
- > Caractéristiques des systèmes répartis
- > Systèmes répartis vs. systèmes centralisés
- > Mise en oeuvre des applications réparties

Introduction à RPC

- > Quelques rappels
- > Présentation de RPC
- > Fonctionnement général de RPC

Le protocole RPC

- > RPC & XDR
- > Le langage RPC
- > Fonctionnement de RPC
- > Gestion des erreurs
- > Données transmises
- > Accès aux serveurs et aux services

2 crédits ECTS

Volume horaire

Cours Magistral : 4h

Travaux Dirigés : 4h

Travaux Pratiques : 8h

Pré-requis nécessaires

Programmation C Avancée

Compétences visées

- > Introduction à RPC : Remote Procedure Call
- > Développement d'une application répartie client/serveur en utilisant le protocole RPC
- > Mise en œuvre de l'application avec l'outil RPCgen (en langage C sous linux).

Bibliographie

- > aux systèmes répartis, Frank Singhoff, UBO, 2011
- > Technologie et architectures Internet, Pierre-Yves Cloux & al., DUNOD, 2002
- > Algorithmes distribués e

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
EC	CC	Travaux Pratiques		1/2	
EC	CT	Ecrit - devoir surveillé	60	1/2	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
EC	CT	Ecrit - devoir surveillé	60	1	pas de report de CC

Réseaux IP : IHM

Présentation

L'accent est mis sur les points suivants :

1. Les flots de données, pré-requis à l'utilisation des sockets pour la communication via des fichiers le réseaux ainsi que la communication série (port RS232, bluetooth, sans fil, etc.)
2. La programmation des sockets en Java (comparaison/différence avec le C),
3. Les threads et la synchronisation. Cette partie est indispensable dans toute application Java. Aussi, elle permet de gérer les accès aux fichiers/ressources (plusieurs utilisateurs accédant en parallèle à une même variable ou via le réseau au même fichier),
4. La création d'une interface utilisateur : nécessite des notions sur la structuration des applications en packages et les IHM avec Swing et Java FX,
5. Introduction à la sérialisation et au protocole de communication MQTT (API Paho de Eclipse)

3 crédits ECTS

Volume horaire

Cours Magistral : 9h

Travaux Dirigés : 9h

Travaux Pratiques : 9h

Objectifs

Cet EC permet d'introduire les concepts de programmation Java permettant de développer des applications complètes intégrant à la fois des aspects orientées réseaux (Web et Sockets) et la conception d'interfaces utilisateur (Swing et JavaFX). Elle doit permettre à un étudiant de développer des applications clients/serveurs de type Chat, client FTP, etc.

Pré-requis nécessaires

programmation Java

Compétences visées

Java réseau et IHM

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
EC	CT	Ecrit - devoir surveillé	60	1/2	
Travaux Pratiques	CC	Travaux Pratiques		1/2	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
EC	Autre modalité	Ecrit - devoir surveillé	60	1	

Bloc transversal S6 IFA-CDA

8 crédits ECTS

Stage (8 semaines)

Présentation

Un stage de 8 semaines minimum complète le dernier semestre de licence informatique. Ce stage (essentiellement en entreprise) doit permettre à l'étudiant d'appliquer les enseignements suivis au cours de l'année.

Le stage est évalué à la fois par le tuteur en entreprise et par la rédaction d'un rapport ainsi qu'une soutenance orale.

4 crédits ECTS

Volume horaire

Stages : 280h

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CC	Travaux Pratiques		1/3	Note de travail en entreprise
UE	CT	Ecrit - rapport		1/3	
UE	CT	Oral - soutenance	15	1/3	

Anglais S6

2 crédits ECTS

Volume horaire

Travaux Dirigés : 16h

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
Travaux Dirigés	CC	Ecrit et/ou Oral		100/100	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
Travaux Dirigés	CT	Oral	15	100/100	

Communication S6

2 crédits ECTS

Volume horaire

Travaux Dirigés : 12h

Modalités de contrôle des connaissances

Session 1 ou session unique - Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CC	Ecrit et/ou Oral		50/100	
UE	CT	Ecrit - devoir surveillé	60	50/100	

Session 2 : Contrôle de connaissances

Nature de l'enseignement	Modalité	Nature	Durée (min.)	Coefficient	Remarques
UE	CT	Ecrit - devoir surveillé	60	100/100	