

INFORMATIQUE 3eme ANNEE (INFO3)

INFORMATION 3rd YEAR

Maquette des enseignements

Semestre : 5

Code Apogée	Intitulé	Etcs.	Coef.	Coef / UE	Vol. (H)
KAX5U001	UE1 : TRONC COMMUN 1 (COMMON CORE PROGRAMME 1)	8	8		
KAX5ANTC	ANGLAIS TC (English CC)			0.40	37
KAX5COTC	COMMUNICATION TC (Communication CC)			0.00	14
KAX5MATC	MATHEMATIQUES TC (Mathematics CC)			0.40	38
KAX5EDTC	ECONOMIE - DROIT TC (Economics - Law CC)			0.20	18
KARI5U02	UE2: MATHEMATIQUES POUR L'INFORMATIQUE (Mathematics for Informatics)	7	7		
KARI5M05	MATHEMATIQUES DISCRETES (MD) (Discrete Mathematics (DM))			0.50	35
KARI5M06	AUTOMATES ET GRAMMAIRES (A&G) (Automata theory and Grammars (A&G))			0.50	36
KARI5U03	UE3: ALGORITHME-PROGRAMMATION 1 (Algorithms and Programming 1)	8	8		
KARI5M07	ALGORITHMIQUE ET PROGRAMMATION IMPERATIVE (Imperative Programming and Algorithmics (AIP))			0.40	57
KARI5M12	PROGRAMMATION ORIENTEE OBJET (POO) (Object Oriented Programming)			0.60	64
KARI5U04	UE4: ARCHITECTURE-SYSTEME 1 (Architecture-System 1)	7	7		
KARI5M10	ARCHITECTURE LOGICIELLE ET MATERIELLE (ALM1) (Hardware and Software Architecture (HSA1))			1.00	62
KARI5M11	LINUX INSTALL PARTY (Linux Install Party)			0.00	7

Semestre : 6

Code Apogée	Intitulé	Etcs.	Coef.	Coef / UE	Vol. (H)
KAX6U001	UE1 : TRONC COMMUN 2 (COMMON CORE PROGRAMME 2)	7	7		
KAX6ANTC	ANGLAIS TC (English CC)			0.30	22
KAX6COTC	COMMUNICATION TC (Communication CC)			0.10	12
KAX6GETC	GESTION TC (Management CC)			0.40	39
KAX6MASP	MATHEMATIQUES DE SPECIALITE ()			0.20	38
KARI6U02	UE2 : MATHEMATIQUES ET APPLICATIONS (Mathematics and applications)	5	5		
KARI6M06	TRAITEMENT DU SIGNAL (TS) (Signal Processing (SP))			0.46	29
KARI6M05	METHODES NUMERIQUES (MN) (Numerical Methods (NM))			0.54	30
KARI6U03	UE3: ALGORITHME-PROGRAMMATION 2 (Algorithms and Programming 2)	6	6		
KARI6M16	PROGRAMMATION WEB (Web Programming)			0.45	32
KARI6M08	ALGORITHMIQUE AVANCEE (Advanced Algorithms (AA))			0.35	26
KARI6M09	LANGAGE C POUR LA PROGRAMMATION SYSTEME (CPS) (C Language for System Programming)			0.20	26
KARI6U04	UE4: ARCHITECTURE-SYSTEME 2 (Architecture-System 2)	6	6		
KARI6M10	COMMUNICATIONS NUMERIQUES (CN) (Numerical Communications (DC))			0.17	27
KARI6M11	RESEAUX (RES) (Networks (NS))			0.33	33
KARI6M12	ARCHITECTURE LOGICIELLE ET MATERIELLE (ALM2) (Hardware and Software Architecture)			0.50	39
KARI6U05	UE5 : PROJETS ET ACTIVITES SPORTIVES (Projects and sport activities)	6	6		
KARI6M13	PROJET LOGICIEL NIVEAU SYSTEME (PLS) (Team Project in System Programming (SPS))			0.25	14
KARI6M14	PROJET LOGICIEL NIVEAU APPLICATIF (PLA) (Software project: the application level (SPA))			0.50	15
KARI6M15	MISE EN SITUATION VIA LE SPORT (Sport and Social Enhancement)			0.25	24

Détail des enseignements

Module : KAX5U001 - UE1 : TRONC COMMUN 1 (COMMON CORE PROGRAMME 1)

Matière : KAX5ANTC - ANGLAIS TC (English CC)

Objectifs

Renforcement des acquis du B1 dans le but d'atteindre le niveau B2
Introduction au discours scientifique
Développement de vocabulaire scientifique
Apprentissage de la rédaction et de la structure d'un rapport scientifique
Ouverture à la communication orale formelle et informelle

Intended learning outcomes

Reinforcing B1 skills in order to reach B2
Introduction to scientific discourse
Development of scientific vocabulary
Learning to write and organise a scientific report
Introduction to formal and informal communication

Pré-requis

Niveau B1 en anglais

Prerequisites

B1 Level

Plan du cours

1 Expression Orale

1.1 Description d'objets

La forme, la dimension, la position, les matériaux, l'utilisation
Causes et conséquences

1.2 Techniques de présentation orale

Structuration, Introduction, Liens, Présenter de l'information visuelle, Conclusion

1.3 Prononciation

Connaissance et pratique des phonèmes anglais

Connaissance et pratique de l'accentuation

Prononciation de chiffres, de lettres et de symboles mathématiques

2 Expression Écrite

2.1 Rédaction de texte descriptif

Utilisation à l'écrit des fonctions apprises en 1.1

2.2 Rédaction de description de données statistiques

2.3 Rédaction de lettre de candidature

Utilisation de tournures standard

2.4 Décrire une situation, une expérience présente et passée.

2.5 Prise de notes

Rédaction de synthèse à partir d'un texte écrit ou oral, ou à la suite d'un échange entre apprenants

2.6 Rédaction de mini-rapport scientifique

En binôme entre deux filières, validé par un jury mixte professeur d'anglais/professeur scientifique

3 Compréhension orale et écrite :

3.1 Compréhension de descriptions et de présentations décrites en 1.1, 1.2 et 1.3

3.2 Compréhension globale de documents audio et vidéo authentiques

3.3 Compréhension d'échanges d'information en face à face

3.4 Compréhension détaillée de textes et de documents audio/vidéo de vulgarisation scientifique

Course content

1 Speaking Skills

1.1 Object Description

Shape, measurement, position, materials, use

Cause and consequences

Description of statistical data

Graph description

Future hypothesis

1.2 Presentation techniques

Structure, Introduction, Signposting, Visuals, Conclusion

1.3 Pronunciation

Awareness and use of English phonemes
Awareness and use of stress patterns
Pronunciation of numbers, letters and mathematical symbols

2 Writing Skills

2.1 Descriptive texts

Written use of functions studied in 1.1

2.2 Written description of statistical data

2.3 Cover letter (use of standard forms)

2.4 Describing a present, past situation or experience

2.5 Note-taking

Summary-writing based on a written or audio document, or following pair or group work

2.6 Writing of short scientific report

In pairs between two different departments, assessed by a combined panel English teacher/Science teacher

3 Listening/Reading Comprehension

3.1 Understanding of descriptions/presentations described above.

3.2 Global understanding of authentic audio/video documents

3.3 Understanding of information exchanges face-to-face or on the telephone

3.4 Detailed understanding of scientific texts and audio/video documents

Bibliographie

4.1 Livres et Ouvrages

- Fascicule de cours de 3ème année
- Upjohn, Jonathan, Minimum Competence in Scientific English, PUG
- Oxford Advanced Learner's Dictionary, OUP
- New Scientist Magazine

4.2 Documents électroniques

- 1) www.newscientist.com
- 2) www.oup.com/elt/oald/
- 3) www.bbc.co.uk

Course literature

4.1 Books and Reference Books

- 3rd year booklet (internal document)
- Upjohn, Jonathan, Minimum Competence in Scientific English, PUG
- Oxford Advanced Learner's Dictionary, OUP
- New Scientist Magazine

4.2 Electronic Resources

- 1) www.newscientist.com
- 2) www.oup.com/elt/oald/
- 3) www.bbc.co.uk

Compétences

Peut comprendre le discours scientifique de base

Peut comprendre un document écrit ou sonore de vulgarisation scientifique

Peut présenter de façon formelle des informations scientifiques et générales

Peut s'exprimer sur des sujets variés et échanger des informations avec plusieurs interlocuteurs

Peut synthétiser et rédiger de façon claire et structurée des informations d'ordre scientifique et générale

Matière : KAX5COTC - COMMUNICATION TC (Communication CC)

Objectifs

-

Acquérir un savoir-faire et une meilleure aisance dans la prise de parole en public : discours structuré et vivant, clair et concis

Prendre conscience des différents paramètres en jeu dans une prestation orale, notamment de la communication non verbale

Gérer efficacement son trac et ses émotions devant un public

Améliorer ses capacités à argumenter, convaincre et écouter.

Intended learning outcomes

-

Acquire know-how and greater fluency in public speaking: structured and lively, clear and concise speech

Become aware of the different parameters involved in oral performance, especially non-verbal communication

Manage emotions in front of an audience

Improve ability to argue, convince and listen

Pré-requis

Aucun

Prerequisites
None
Plan du cours
7 séances thématiques - Fondamentaux de la communication relationnelle - Esprit d'équipe - Communiquer en groupe - Valorisation - Improvisation et sens de la répartie - Communication non verbale - Gestion du stress
Course content
7 thematic sessions - Fundamentals of relational communication - Team spirit - Communicate in a group - Valuation - Improvisation and sense of repartee - Non-verbal communication - Stress management
Bibliographie
- Différents ouvrages de communication donnés dans le cadre du Tronc commun
Course literature
- Different communication works given in the framework of the Common Core

Compétences
Good in written and oral communication Control of gestures

Matière : KAX5MATC - MATHEMATIQUES TC (Mathematics CC)

Objectifs

Objectif des mathématiques générales de début d'année : acquérir ou conforter les notions de base en mathématiques : équations différentielles, nombres complexes, intégration, algèbre linéaire.

Intended learning outcomes

The main objective of this course is to acquire or reinforce the basic notions of mathematics: differential equations, complex numbers, integration, linear algebra

Pré-requis

Fonctions usuelles, éléments de calcul vectoriel, calcul intégral élémentaire.

Développements limités, équations différentielles linéaires, calcul intégral, intégrales impropres et séries, séries entières.

Prerequisites

Usual functions, vectorial calculus, elementary integral calculus.

Taylor expansion, linear differential equations, integral calculus, improper integrals, numerical series

Plan du cours

MATHEMATIQUES GENERALES

1. Analyse

Nombres complexes

Étude de fonctions

Développements limités

Équations différentielles linéaires

Calcul intégral

Intégrales impropres

2. Algèbre linéaire

Opérations élémentaires sur une matrice rectangulaire

Algorithme de Gauss et applications

Inversion d'une matrice et calcul de déterminant

Diagonalisation d'une matrice

Course content

GENERAL MATHEMATICS

1. Analysis

Complex numbers

Study of a function

Taylor expansion
Differential equations
Integrals
Generalized integrals

2. Linear algebra

Basic operations on a rectangular matrix
Gauss algorithm and applications
Inversion of a matrix and calculus of a determinant
Matrices diagonalization

Bibliographie

mathématiques générales : Algèbre et analyse, Thuillier, Ed. Belloc.

Course literature

mathematiques generales : Algebre et analyse, Thuillier, Ed. Belloc.

Compétences

Savoir appliquer des connaissances en mathématique pour résoudre des problèmes complexes

Know how to apply mathematical knowledge to solve complex problems

Matière : KAX5EDTC - ECONOMIE - DROIT TC (Economics - Law CC)

Objectifs

Acquérir une culture générale sur les principales problématiques en économie et en droit.
Poser un regard éclairé sur l'actualité économique et sur les rôles et droits du citoyen.

Intended learning outcomes

Acquire a general knowledge of the main issues in economics and law.
Take an enlightened look at current economic events and the roles and rights of the citizen.

Pré-requis

Aucun

Prerequisites

None

Plan du cours

Economie :
- Le circuit économique
- La croissance
- Emploi et chômage

Droit :
- L'ordre judiciaire français
- Les sources du droit
- La personnalité juridique et les droits subjectifs
- Les contrats

Course content

Economics:
- The economic cycle
- economic growth
- Employment and unemployment

Law :
- The French Judicial system
- Law sources
- Legal personality and subjective rights
- Contracts

Bibliographie

Introduction au droit et thèmes fondamentaux du droit civil
J.L. Aubert, E. Savaux
Editions Broché

Course literature

Introduction au droit et thÃ©mes fondamentaux du droit civil
J.L. Aubert, E. Savaux
Editions BrochÃ©

Compétences

Savoir appliquer les connaissances en économie et en droit dans l'entreprise

Module : KARI5U02 - UE2: MATHEMATIQUES POUR L'INFORMATIQUE (Mathematics for Informatics)

Matière : KARI5M05 - MATHEMATIQUES DISCRETES (MD) (Discrete Mathematics (DM))

Objectifs

Le but de l'enseignement de MATHEMATIQUES DISCRETES est l'approfondissement de plusieurs notions fondamentales de mathématiques, accompagné d'un aperçu de leur utilisation en informatique. Ce cours est conçu comme une introduction aux études de INFO : on y étudie un petit nombre de concepts-clés (notions d'algorithme, de langage, de codage,...) qui sous-tendent l'étude des outils informatiques vue dans d'autres cours.

1. Les ensembles et leurs partitions sont illustrés par des exemples portant sur les mots et les langages.
2. Les fonctions et leurs images réciproques fournissent un point de vue unifié sur les problèmes de codage : encodage binaire, compression, cryptographie, codes correcteurs.
3. Les relations d'équivalence et la notion de passage au quotient permettent d'appréhender des questions fondamentales de sémantique des langages de programmation.

Intended learning outcomes

The aim of the Discrete Mathematics course is to explore several fundamental notions from mathematics and see how they apply to computer science. This course is devised as an introduction to the contents of the INFO curriculum: we study a small set of key concepts (the notions of algorithm, language, coding,...) that underly many of the tools specific to computer science and studied in other courses.

1. Sets and their partitions are illustrated with examples about words and languages.
2. Functions and their inverse images yield a unified point of view over coding problems: binary encoding, compression, cryptography, error correction.
3. Equivalence relations and quotients give a general framework for studying the issues of semantics in programming languages.

Pré-requis

Aucun pré-requis

Prerequisites

No prerequisites

Plan du cours

1. Ensembles et partitions. Mots et langages.
2. Fonctions et images réciproques. Quelques problèmes de codage.
3. Relations d'équivalence et passage au quotient. Notions de sémantique.

Course content

1. Sets and partitions. Words and langages.
2. Functions and inverse images. Some coding theory.
3. Equivalence relations and quotients. Some semantic issues.

Bibliographie

Jacques Vélou. Méthodes mathématiques pour l'informatique. Dunod, Collection Sciences Sup, 4e édition (2005).

Compétences

- Maîtriser les outils mathématiques de base que sont les ensembles, fonctions et relations d'équivalence.
- Savoir trouver les bons arguments pour justifier une affirmation, sans forcément avoir recours à un formalisme mathématique sophistiqué.
- Penser aux algorithmes en termes de "contrats" : inclure une spécification et des tests en complément du code.

Matière : KARI5M06 - AUTOMATES ET GRAMMAIRES (A&G) (Automata theory and Grammars (A&G))

Objectifs

L'enseignement "AUTOMATES et GRAMMAIRES" comporte 4 volets :

1. La présentation de résultats fondamentaux de l'informatique : une technique de preuve de correction de programmes, différents modèles de calculs (séquentiel, parallèle, non-déterministe), la réalisation en un temps fini d'opérations sur des données infinies (les langages représentés sous forme d'automates), la récursivité
2. L'illustration des ces notions dans le cas des automates et des grammaires
3. Un exemple concret d'utilisation récente des automates : vérification de drivers, protocoles médicaux, personnage autonome dans les jeux vidéo, ...
4. Un exemple concret d'utilisation des grammaires attribuées
5. La présentation d'un résultat surprenant : les automates d'arbres, l'ordinateur quantique, les limites de l'informatique, ...

Intended learning outcomes

The course "AUTOMATES and GRAMMARS" has 4 components:

1. The presentation of fundamental results of computer science: a technique of correctness proof of programs, different computation models (sequential, parallel, non-deterministic), the computation in finite time of operations on infinite data (the languages represented as automata), recursion
2. The illustration of these notions in the case of automata and grammars
3. A concrete example of recent use of automata: verification of drivers, medical protocols, autonomous character in video games, ...
4. A concrete example of the use of attributed grammars
5. The presentation of a surprising result: tree automatons, the quantum computer, the limits of computing, ...

TARGETED SKILLS

- know how to prove that a program is correct
- know how to program using an automaton
- know how to write a simple analyzer / translator

Pré-requis

Connaître les rudiments d'un langage impératif tel que C

Prerequisites

Basic knowledge of an imperative language such as C

Plan du cours

1. Preuve de correction partielle de programmes par la technique de Floyd-Hoare-Dijkstra : comment être sûr que un programme fait bien ce qu'on attend. Autrement dit, monterez-vous dans l'avion dont vous avez programmé le pilote automatique ?
2. Automates (à nombres d'états finis/à pile, déterministes/non-déterministes) : quel est le modèle de calcul d'un processeur ? y'a t'il des langages (des modèles) plus puissants que d'autres ?
3. Représentations équivalentes (des grammaires régulières aux équations d'Arden aux expressions régulières et finalement jusqu'aux automates à états finis) : comment passent-on d'une description lisible à une version utilisable par un processeur ?
4. Application et implantation des automates : les automates sont utilisés pour programmer (des analyseurs lexicaux, des micro-contrôleurs, des interfaces, des protocoles, des jeux,...), pour piloter (des chaînes de production, des systèmes cyber-physiques,...) et pour vérifier (des drivers, des politiques de sécurité ...) en fait on en trouve un peu partout.
5. Grammaires attribuées et génératives : les grammaires sont le quotidien des informaticiens qui ne cessent de traduire un langage ou des données d'un format vers un autre.
6. Application : implantation en ANTLR ou JavaCC d'un interpréteur d'expressions booléennes

Course content

1. Proof of partial program correctness by the Floyd-Hoare-Dijkstra technique: how to be sure that a program is doing what is expected. In other words, will you get on the plane which runs an autopilot you programmed?
2. Automata (with finite state / stack numbers, deterministic / non-deterministic): what is the computation model of a processor? are there languages more powerful than others?
3. Equivalent representations (from Regular Grammars to Arden's Equations then to Regular Expressions and finally to Finite State Automata): how do we go from a readable description to a version usable by a processor?
4. Application and implementation of Automata (ie. Programmable Logic Controller): PLC are used to program (lexical analyzers, micro-controllers, interfaces, protocols, games, ...), to control (production chains, physical, ...) and to check (drivers, security policies ...) in fact we find them everywhere.
5. Attributed and Generative Grammars: Grammars are the everyday tool of developers who translate language or data from one format to another.
6. Application: implementation in ANTLR or JavaCC of an interpreter of Boolean expressions

Bibliographie

1. Introduction à la calculabilité, Pierre Wolper, Éditions Dunod (3eme édition, 2006)
2. cours en ligne sur le site web A&G : <http://www-verimag.imag.fr/~perin/enseignement/RICM3/infraeg/>

Compétences

COMPÉTENCES VISÉES

- savoir prouver qu'un programme est correct
- savoir programmer à l'aide d'automate
- savoir écrire un analyseur/traducteur simple

Module : KARI5U03 - UE3: ALGORITHME-PROGRAMMATION 1 (Algorithms and Programmation 1)

Matière : KARI5M07 - ALGORITHMIQUE ET PROGRAMMATION IMPERATIVE (Imperative Programming and Algorithmics (AIP))

Objectifs

L'objectif du cours est d'introduire les bases de l'algorithmique dans un cadre de programmation impérative. Par delà la présentation des algorithmes, structures de données et méthodologies de programmation classiques, le cours insistera sur les notions d'invariant et de complexité, fondamentales pour le développement d'applications à la fois robustes et performantes. Chaque nouvelle notion sera illustré par de nombreux exemples explicatifs. Les cours sont accompagnés de travaux pratiques de programmation et de travaux dirigés au cours desquels les étudiants apprennent à créer leurs propres algorithmes de manière robuste et performante.

Intended learning outcomes

In this course we introduce the basics of Imperative Programming and Algorithmics.

Pré-requis

néant

Prerequisites

None

Plan du cours

Notions d'états, d'assertions d'invariants. Illustration sur l'algorithme du tri par insertion

Introduction à un langage de programmation

Notions de complexité, illustration sur le tri par insertion

Programmation par module/paquetage

Algorithmes basés sur des modèles séquentiels

Structures de données pour les ensembles dynamiques

Pointeurs

Tables de hachage

Arbres

Course content

Notions of states, assertions and invariants. Illustration with the insertion sort algorithm.

Introduction to a programming language

Notion of complexity, illustration with the insertion sort

Programming with modules/packages

Sequential models of algorithms

Data structures for dynamic sets

Pointers

Hash tables

Trees

Bibliographie
Algorithmique, Cours avec 957 exercices et 158 problèmes, Thomas Cormen, Charles Leiserson, Ronald Rivest, éditions Dunod

Course literature
Thomas Cormen, Charles Leiserson, Ronald Rivest, "Introduction to Algorithms (3rd edition), MIT-Press

Matière : KARI5M12 - PROGRAMMATION ORIENTEE OBJET (POO) (Object Oriented Programming)
--

Objectifs
Ce cours revient sur la programmation impérative et ses concepts pour faire le contraste avec la programmation orientée-objet. Pour cela, nous utilisons le langage de programmation Java qui permet de faire à la fois de la programmation impérative et orientée-objet. Nous abordons les concepts du langage Java mais aussi les techniques et les outils de programmation. Les connaissances et l'expérience pratique acquises seront utiles en développement logiciel au delà du seul langage de programmation Java, quel que soit le langage et les outils utilisés.

Intended learning outcomes
This course revisits imperative programming in order to contrast it with object-oriented programming. To that end, we use the Java programming language since it is a language that permits both imperative and object-oriented programming. We will discuss the programming concepts of the language but also coding techniques and tools. The skills and know-how acquired will be useful in software development beyond the only Java programming language, regardless of the language and tools used.

Pré-requis
Aucun

Prerequisites
None

Plan du cours
Ce cours est organisé en "sprints", avec plusieurs tâches par sprint. Le cours suit une progression en trois phases, guidant l'étudiant vers une meilleure maîtrise du développement logiciel.
<ol style="list-style-type: none"> 1. La première phase est sur la programmation impérative et les outils d'aide à la programmation. Le but est d'établir une fondation commune aux étudiants venant d'horizons divers. 2. La seconde phase est la découverte de la programmation orientée-objet, au travers des concepts de ?urs d'objets et de classes. Le focus est sur le graphe d'objets et l'exécution orientée-objet. 3. La troisième phase est l'introduction du polymorphisme et de l'héritage, découvrant la puissance de la programmation orientée-objet et des outils associés.

Course content
This course is organized in "sprints", with several tasks per sprint. The course follows a progression in three incremental phases, taking all students on a journey towards mastering software development better.
<ol style="list-style-type: none"> 1. The first phase is about imperative programming and programming tools, setting up a common foundation across students coming from backgrounds. 2. The second phase is about discovering object-oriented programming, understanding the core concepts of objects and classes. The focus is on the object graph and object-oriented execution. 3. The third phase is about introducing polymorphism and inheritance, discovering the power of object-oriented programming and tools.

Bibliographie
<ol style="list-style-type: none"> 1. B.Meyer, Object Oriented Software Construction, Second Edition. Prentice Hall, 1997 2. G.Masini, A.Napoli, D.Colnet, D.Léonard, K.Tombre, Les Langages à objets, InterÉditions 1989 3. Bruce Eckel, Thinking in Java, 4th Edition Prentice-Hall PTR, 2006.

Course literature
<ol style="list-style-type: none"> 1. B.Meyer, Object Oriented Software Construction, Second Edition. Prentice Hall, 1997 2. G.Masini, A.Napoli, D.Colnet, D.Léonard, K.Tombre, Les Langages à objets, InterÉditions 1989 3. Bruce Eckel, Thinking in Java, 4th Edition Prentice-Hall PTR, 2006.

Module : KARI5U04 - UE4: ARCHITECTURE-SYSTEME 1 (Architecture-System 1)
--

Matière : KARI5M10 - ARCHITECTURE LOGICIELLE ET MATERIELLE (ALM1) (Hardware and Software Architecture (HSA1))
--

Objectifs
Cet enseignement permet aux futurs ingénieurs informaticiens d'avoir un idée précise de l'architecture d'un ordinateur et de l'exécution des programmes à l'aide de ses composants matériels. Ces notions permettent d'aborder sereinement des domaines tels que la programmation en langage évolué, la compilation de ces programmes, les systèmes d'exploitation d'un ordinateur, la conception de circuits.

Intended learning outcomes
This teaching allows future computer engineers to have a clear idea of the architecture of a computer and the execution of programs using its hardware components. These concepts make it possible to serenely address areas such as advanced language programming, the compilation of these programs, the operating systems of a computer, the design of circuits.

Pré-requis
Notions élémentaires d'algorithmique

Prerequisites
Basic notions of algorithmic

Plan du cours
Cet enseignement présente d'une part les composants électroniques élémentaires permettant de comprendre l'architecture des processeurs et d'autre part la programmation en langage machine et d'assemblage. Il permet ainsi de comprendre précisément l'exécution d'instructions machines par le processeur.
Une deuxième partie (voir ALM2) aborde ensuite les périphériques, les mécanismes d'entrée/sortie du processeur, les interruptions extérieures des programmes et les notions élémentaires à la base des systèmes d'exploitations.

Course content

This teaching presents on the one hand the elementary electronic components making it possible to understand the architecture of the processors and on the other hand programming in machine language and assembly. It allows to understand precisely the execution of machine instructions by the processor.

A second part (see ALM2) then discusses peripherals, processor input / output mechanisms, external program interrupts, and the basics of operating systems.

Bibliographie

Architectures Logicielles et Matérielles : Cours, études de cas et exercices corrigés, P. Amblard, J.C. Fernandez, F. Lagnier, F. Maraninchi, P. Sicard, P. Waille, Editions DUNOD 2000

Compétences

À l'issue de ce cours les étudiants sont capables :

- de concevoir et assembler les différents composants d'un processeur (registres, bus, unité arithmétique...),
- de maîtriser précisément le processus d'exécution d'une instruction machine (se trouvant en mémoire) par un processeur;
- de comprendre précisément les notions de variables dans un programme et d'adresse en mémoire de ces variables, à partir de là, de maîtriser la notion de pointeur utilisée dans les langages de programmation tel que le C,
- d'écrire des programmes simples en C utilisant ces notions,
- d'écrire des programmes mélangeant du C et de l'assembleur ARM,
- de comprendre le résultat de la compilation en assembleur (ARM) d'un programme C.

Matière : KARI5M11 - LINUX INSTALL PARTY (Linux Install Party)

Objectifs

Installer sur leur machine le système d'exploitation linux et les outils de programmation qui leur seront utiles en année 3.

- À l'issue de leur formation les ingénieurs RICM seront capables de gérer un parc de machines.
- La L.I.P est une première étape vers cet objectif qui amène les élèves à être capable d'administrer leur propre machine de travail

Intended learning outcomes

Install on their machine the linux operating system and the programming tools that will be useful for them in year 3.

- Upon completion of their training RICM engineers will be able to manage a fleet of machines.
- The L.I.P is a first step towards this goal that leads students to be able to administer their own work machine

Pré-requis

On demande aux élèves de la filière INFO de s'équiper d'une machine portable personnelle afin d'effectuer les nombreux travaux pratiques et projets de programmation.

Prerequisites

Students of the INFO program are asked to get a personal portable machine in order to perform the many practical work and programming projects.

Plan du cours

À leur arrivée, les nouveaux entrants en RICM sont conviés à une Linux Install Party (L.I.P)

- Présentation des installations possibles : machine virtuelle ou native avec dual boot, organisation des disques et partitions (1h)
- Installation du système d'exploitation LINUX (3h)
- Introduction à l'administration de la machine (1h)
partitionnement, hiérarchie de répertoires, gestion des permissions, installation de paquet, création d'utilisateur

Course content

Upon arrival, RICM students are invited to a Linux Install Party (LIP)

- Presentation of two ways of installing Linux : virtual machine or native with dual boot, management of disks and partitions (1h)
- Installation of the LINUX operating system (3h)
- Introduction to machine administration (1h) :
partitioning, hierarchy of directories, permissions management , package installation, user creation

Bibliographie

"Installer Linux" de S.Desreux & E.Cornet, éditions H&K

Module : KAX6U001 - UE1 : TRONC COMMUN 2 (COMMON CORE PROGRAMME 2)

Matière : KAX6ANTC - ANGLAIS TC (English CC)

Objectifs

Renforcement des compétences B1 pour tendre vers le niveau B2

Introduction au discours scientifique

Développement de vocabulaire scientifique

Rédaction d'un rapport scientifique (introduction à la recherche)

Ouverture à la communication orale formelle et informelle

Intended learning outcomes

Reinforcing B1 skills in order to reach B2

Introduction to scientific discourse

Development of scientific vocabulary

Learning to write and organise a scientific report

Pré-requis

Niveau B1 en anglais

Prerequisites

B1 Level

Plan du cours

1.1 Exprimer le futur

- 1) Les différentes formes du futur
- 2) Exprimer la probabilité et utiliser les modaux

1.2 Description de données statistiques graphiques

- 1) Causes et conséquences
- 2) Hypothèses futures

1.3 Techniques de présentation orale

- 1) Introduction
- 2) Liens
- 3) Présenter de l'information visuelle
- 4) Conclusion

1.4 Prononciation

- 1) Connaissance et pratique des phonèmes anglais
- 2) Connaissance et pratique de l'accentuation

2. Expression Ecrite

2.1 Rédaction de texte descriptif

Utilisation à l'écrit des fonctions apprises au S5
Rédaction de mini-rapport scientifique

2.2 Rédaction de description de données statistiques

Utilisation à l'écrit des fonctions appris en 1.2

2.3 Prise de notes

Rédaction de résumé à partir d'un texte écrit ou oral, ou à la suite d'une conversation

3. Compréhension Orale et écrite :

3.1 Compréhension de descriptions et de présentations décrites en 1.1, 1.2 et 1.3

3.2 Compréhension globale de documents audio et vidéo authentiques

3.3 Compréhension d'échanges d'information en face à face ou au téléphone

3.4 Compréhension détaillée de textes généraux et de vulgarisation scientifique

Course content

1.1 Expressing the future

- 1) Different ways to express the future
- 2) Expressing probability & using modal verbs

1.2 Describing trends (statistical data / graphs)

- 1) Causes and consequences
- 2) Future Hypotheses

1.3 Presentation Skills

- 1) Introduction
- 2) Signposting
- 3) Presenting visual data
- 4) Conclusion

1.4 Pronunciation

Awareness and use of English phonemes
Awareness and use of stress patterns

2 Written Production

2.1 Writing a report

Using Semester 5 written phrases and skills in order to write a mini scientific report

2.2 Writing about trends

Analysing statistics and graphs as learnt in 1.2 in an essay

2.3 Note-taking

Writing a summary from notes taken from a written document or an audio document, or a conversation

3. Reading and Listening Comprehension

3.1 Understanding descriptions and presentations as described in 1.1, 1.2 and 1.3

3.2 Global comprehension of authentic audio and video recordings

3.4 Detailed comprehension of general and scientific texts

Bibliographie

4.1 Livres et Ouvrages

- Fascicule de cours de 3ème année

- Upjohn, Jonathan, Minimum Competence in Scientific English, PUG

- Oxford Advanced Learner's Dictionary, OUP

-New Scientist Magazine

4.2 Documents électroniques

1) www.newscientist.com

2) www.oup.com/elt/oald/

3) www.bbc.co.uk

Course literature

4.1 Books and Reference Books

- 3rd year booklet (internal document)

- Upjohn, Jonathan, Minimum Competence in Scientific English, PUG

- Oxford Advanced Learner's Dictionary, OUP

-New Scientist Magazine

4.2 Electronic Resources

1) www.newscientist.com

2) www.oup.com/elt/oald/

3) www.bbc.co.uk

Compétences

Peut comprendre le discours scientifique de base

Peut comprendre un document écrit ou sonore de vulgarisation scientifique

Peut présenter de façon formelle des informations scientifiques et générales

Peut s'exprimer sur des sujets variés et échanger des informations avec plusieurs interlocuteurs

Peut synthétiser et rédiger de façon claire et structurée des informations d'ordre scientifique et générale

Matière : KAX6COTC - COMMUNICATION TC (Communication CC)

Objectifs

améliorer les compétences rédactionnelles ; optimiser une prise de notes en vue d'une synthèse ; élaborer une argumentation avec des outils linguistiques et rhétoriques ; s'initier à quelques formes de documents professionnels ; réviser les principales bases de l'orthographe en vue d'une relecture d'un document professionnel

Intended learning outcomes

argumentation with linguistic and rhetorical tools; introduction to some forms of professional documents; review the main bases of spelling with a view to proofreading a professional document

Pré-requis

compréhension du français courant d'un document audio / vidéo / écrit

Prerequisites

understanding of fluent French from an audio / video / written document

Plan du cours

séance 1 : initiation mindmap ;

séance 2 : test d'orthographe et corrigé ;

séance 3 : sensibilisation aux synonymes et application sur mails professionnels ;

séance 4 : notion de plan et rédaction d'un document type note de service/phase de projet ;

séance 5 : types d'argumentation et rédaction d'un discours d'entreprise ;

séance 6 : examen et autoévaluation

Course content

session 1: mindmap initiation;

session 2: spelling test and answers;

session 3: synonym awareness and application to professional emails;

session 4: concept of a plan and drafting of a standard document memorandum/project phase;

session 5: types of argumentation and writing a company speech;

session 6: examination and self-assessment

Bibliographie

ouvrages et sites donnés par chaque formateur en cours

Course literature
books and sites given by each teacher

Compétences
Maîtriser et gérer sa communication en public

Matière : KAX6GETC - GESTION TC (Management CC)
--

Objectifs

L'objectif du cours est d'acquérir ou consolider des connaissances sur l'entreprise.
L'apprentissage conduira à une compréhension du fonctionnement général des organisations, une connaissance du langage des managers et des gestionnaires, et une préparation au travail en équipe et à l'encadrement, à travers une réflexion sur les pratiques de management.

Intended learning outcomes

The objective of the course is to acquire or consolidate knowledge about the firm.
Learning will lead to understand the general functioning of organizations, to know the language of managers, and to be prepared to teamwork and coaching, through a reflection on management practices.

Pré-requis

cours d'économie et de droit du semestre 5

Prerequisites

economics and law course (semester 5)

Plan du cours

1. Qu'est-ce que l'entreprise ? Définition, réalité économique et apparence juridique, études de cas.
 2. Comprendre l'environnement de l'entreprise. Approche orientée business ; analyse stratégique ; approche orientée marché : marketing ; approche orientée parties prenantes : RSE.
 3. L'organisation de l'entreprise. Les structures organisationnelles, le pouvoir, l'exercice du leadership.
 4. Les ressources mises en oeuvre dans l'activité économique. Ressources humaines, matérielles, et financières.
 5. Le déploiement des activités. La fonction approvisionnement ; la gestion de production.
 6. La comptabilité financière. Le système d'information comptable, les comptes annuels.
 7. La comptabilité de gestion. Le seuil de rentabilité, les coûts complets.
- Simulation de gestion.

Course content

1. What is the firm ? Definition, economic reality and legal appearance, case studies.
 2. Understand the business environment. Strategic analysis, marketing, stakeholders and CSR.
 3. The organization of the firm. Organizational structures, power, leadership.
 4. The resources used in the economic activity. Human, material and financial resources.
 5. The deployment of activities. The supply function, production management.
 6. Financial accounting. The accounting information system, the annual accounts.
 7. Management Accounting. The breakeven point, the full costs.
- Serious business game.

Bibliographie

Poly de cours + documents complémentaires sur plateforme numérique
Management d'entreprise 360° - Principes et outils de la gestion d'entreprise, de Thomas Durand, éditions Dunod
Comprendre l'entreprise : théorie, gestion, relations sociales - Tony Alberto et Pascal Combemale, Ed. Nathan

Course literature

course documents on numeric plateforme.

Compétences

Compétence en gestion et en droit des affaires

Matière : KAX6MASP - MATHEMATIQUES DE SPECIALITE ()

Objectifs

L'analyse de Fourier et les probabilités doivent permettre de manipuler les outils mathématiques indispensables à d'autres sciences de l'ingénieur : l'analyse de Fourier est indispensable au traitement du signal et à la résolution des équations aux dérivées partielles, les probabilités aux statistiques et au traitement de données.

Intended learning outcomes

Fourier analysis and probabilities allow to manipulate the mathematical tools essential to other engineering sciences: Fourier analysis is essential for signal processing and solving partial differential equations, probabilities for statistics and data processing.

Pré-requis

Calcul intégral, séries, calcul différentiel, probabilité de base.

Prerequisites

Integral calculus, series, differential calculus, elementary probability theory.

Plan du cours

ANALYSE de FOURIER

Série de Fourier d'une fonction périodique L2 sur sa période. Théorème de Parseval
Série de Fourier d'une fonction périodique L1 sur sa période. Théorème de Dirichlet

Premières propriétés de la transformée de Fourier dans L1
Théorème d'inversion de la transformée de Fourier dans L1
Théorème de Plancherel
Convolution et transformée de Fourier

PROBABILITÉS

Espaces probabilisés
Probabilité conditionnelle et indépendance
Généralités sur les variables aléatoires
Variables aléatoires discrètes
Variables aléatoires continues
Fonction caractéristique d'une variable aléatoire
Théorème limite centrale

Course content

FOURIER ANALYSIS

1. Fourier series
Fourier series of a periodic function and Parseval theorem
Fourier series of a periodic function and Dirichlet theorem

2. Fourier transform
Fourier transform basic properties
Fourier transform inversion theorem
Plancherel theorem
Fourier transform and convolution

PROBABILITY

Conditional probability and independence
Discrete random variables
Continuous random variables
Characteristic function of a random variable
Central limit theorem

Bibliographie

analyse de Fourier: Spiegel, Murray Ed. Schaum

probabilités :Vigneron, Logak ; Ed. Diderot

exercices de probabilités: licence, maîtrise et écoles d'ingénieurs(Cottrell...chez Cassini)

Course literature

analyse de Fourier: Spiegel, Murray Ed. Schaum

probabilites :Vigneron, Logak ; Ed. Diderot

exercices de probabilites: licence, maitrise et ecoles d'ingenieurs(Cottrell...chez Cassini)

Compétences

Savoir appliquer des connaissances en mathématique pour résoudre des problèmes complexes

Know how to apply mathematical knowledge to solve complex problems

Module : KARI6U02 - UE2 : MATHEMATIQUES ET APPLICATIONS (Mathematics and applications)

Matière : KARI6M06 - TRAITEMENT DU SIGNAL (TS) (Signal Processing (SP))

Objectifs

Introduction à la théorie du signal et au traitement numérique du signal. Ce cours est nécessaire pour le traitement, le codage et la transmission de l'information. Savoir manipuler et traiter des signaux simples.

Intended learning outcomes
Introduction to signal theory and digital signal processing. This course is necessary for the processing, coding and transmission of information. Ability to handle and process simple signals.
Pré-requis
Calcul intégral élémentaire, analyse de Fourier
Prerequisites
Elementary integral calculus, Fourier analysis
Plan du cours
1 Introduction et bases théoriques 2 Signaux déterministes à temps continu 3 Echantillonnage et quantification des signaux 4 Signaux déterministes à temps discret
Course content
1 Introduction and theoretical basis 2 Continuous-time deterministic signals 3 Sampling and quantization of signals 4 Discrete-time deterministic signals
Bibliographie
-Coulon F., Théorie et traitement des signaux -Kunt M., Traitement numérique des signaux -Delmas J.P., Eléments de théorie du signal : les signaux déterministes -Van Den Eenden A.W.M. et Verhoeckx N.A.M., Traitement numérique du signal

Matière : KARI6M05 - METHODES NUMERIQUES (MN) (Numerical Methods (NM))
Objectifs
L'objectif de ce cours est de présenter des algorithmes numériques, leurs mises en oeuvre sur des processeurs modernes (multicoeurs). Nous nous intéresserons principalement à des algorithmes classiques de l'algèbre linéaire. Les performances de ces algorithmes numériques seront également étudiées, en termes d'opérations flottantes par seconde et d'utilisation de la hiérarchie mémoire.
Intended learning outcomes
The course presents numerical algorithms, their implementations on modern processors (multicore). We will focus mainly on classical algorithms of linear algebra. The performances of these numerical algorithms will also be studied, in terms of floating operations per second and use of the memory hierarchy.
Pré-requis
Mathématiques (analyse et algèbre) niveau L2, calcul matriciel vu dans le enseignement de tronc commun du semestre 5
Prerequisites
L2 level in Mathematics (analysis and algebra), matrix calculus seen in the core curriculum of Semester 5
Plan du cours
- Calcul sur des polynômes - Calcul matriciel et vectoriel (structures pleines et creuses) - Bibliothèque BLAS (Basic Linear Algebra Subprograms) - Evaluation de performance (Calcul Flottant par Seconde) - Parallélisation d'algorithmes numériques (directives OpenMP) - Unités de calcul vectoriel, vectorisation (MMX, SSE, AVX)
3 TPs sont à réaliser: - TP1 sur des fonctions de calcul de polynôme - TP2 sur une bibliothèque d'algèbre linéaire (un sous ensemble de la bibliothèque BLAS) - TP3 sur la parallélisation OpenMP et la vectorisation de la bibliothèque d'algèbre linéaire.
Course content
- Computations on polynomials - Matrix and vector computations (sparse and dense structures) - BLAS Library (Basic Linear Algebra Subprograms) - Performance evaluation (Floating-point Operation per Second) - Parallelizing of numerical algorithms (OpenMP directives) - Vector computation units, vectorization (MMX, SSE, AVX)
3 TPs are realized: - TP1 on polynomial computations - TP2 on a library of linear algebra (a subset of BLAS Library) - TP3 on OpenMP parallelization and vectorization of the library of linear algebra.
Bibliographie
Alfio Quarteroni, Ricardo Sacco, Fausto Saleri: Méthodes Numériques: Algorithmes, analyse et applications, Springer, 2007

Module : KARI6U03 - UE3: ALGORITHME-PROGRAMMATION 2 (Algorithms and Programmation 2)

Matière : KARI6M16 - PROGRAMMATION WEB (Web Programming)

Objectifs
Utiliser les technologies web pour récupérer, traiter et diffuser des données publiques et les mettre en forme.
Intended learning outcomes
Use web technologies to recover, process and disseminate public data and shape them.
Pré-requis
<ul style="list-style-type: none"> - Connaissance de base en programmation imperative - Maîtrise de la ligne de commande unix/linux.
Prerequisites
<ul style="list-style-type: none"> - Basic knowledge in imperative programming - Basic knowledge of the unix / linux command line.
Plan du cours
<ol style="list-style-type: none"> 1. Introduction aux technologies web <ul style="list-style-type: none"> - infrastructure serveur web classique - HTTP (méthodes GET, POST, en-têtes, codes diagnostic) - bases de HTML 5 - introduction à PHP 2. Formats de données <ul style="list-style-type: none"> - manipulation de tableaux en PHP - données structurées en PHP - CSV, prototypage en filtres unix - JSON - XML et validation de données 3. Webservices <ul style="list-style-type: none"> - APIs web et méthodes HTTP - principes REST - exploitation de curl et libcurl 4. Métadonnées <ul style="list-style-type: none"> - notions de multimédia en HTTP/HTML - images - audio et vidéo en HTML5 - métadonnées multimédia - notions de web sémantique et micro-formats - application XML : Atom et RSS
Course content
<ol style="list-style-type: none"> 1. Introduction to web technologies <ul style="list-style-type: none"> - standard web server infrastructure - HTTP (GET, POST methods, headers, diagnostic codes) - HTML 5 basics - introduction to PHP 2. Data Formats <ul style="list-style-type: none"> - manipulation of tables in PHP - structured data in PHP - CSV, prototyping in unix filters - JSON - XML and data validation 3. Webservices <ul style="list-style-type: none"> - Web APIs and HTTP methods - REST principles - exploitation of curl and libcurl 4. Metadata <ul style="list-style-type: none"> - notions of multimedia in HTTP / HTML - images - audio and video in HTML5 - multimedia metadata - notions of semantic web and micro-formats - XML application: Atom and RSS
Bibliographie
<p>Le cours est donné par deux intervenants extérieurs :</p> <ul style="list-style-type: none"> - Sébastien PITTION, ingénieur développeur web chez Viseo - Guillaume ALLÈGRE, administrateur système et réseau chez Silecs

Compétences

Pratique de JavaScript/TypeScript, HTML et CSS/Sass

Matière : KARI6M08 - ALGORITHMIQUE AVANCEE (Advanced Algorithms (AA))

Objectifs

L'objectif de ce cours est de présenter les algorithmes de base pour les structures de type arbre et graphes. Différents types d'arbres seront étudiés (ABR, AVL, A234). Plusieurs types de graphes seront également étudiés (orientés, pondérés). Certains de ces algorithmes seront mis en oeuvre sur 3 TPs. Ces TPs sur ces structures de données permettent aux étudiants de se familiariser avec les mécanismes dynamiques d'allocation mémoire.

Intended learning outcomes

The course presents the basic algorithms for tree and graph structures. Different types of trees will be studied (ABR, AVL, A234). Several types of graphs will also be studied (oriented, weighted). Some of these algorithms will be implemented on 3 TPs. These TPs on these data structures allow students to become familiar with the dynamic mechanisms of memory allocation.

Pré-requis

Bases de l'algorithmique et de la programmation impérative

Prerequisites

Basics on algorithmics and imperative programming

Plan du cours

- Arbre ABR
- Arbre AVL
- Arbre 234
- Arbres bicolore
- Introduction aux graphes
- Parcours en largeur et en profondeur
- Tri topologique
- Algorithme du plus court chemin
- Composantes connexes

Course content

- ABR tree
- AVL tree
- Tree 234
- Two-color trees
- Introduction to graphs
- width and depth traversal
- Topological sorting algorithm
- shortest path algorithm
- strongly connected components

Bibliographie

- D. Beauquier, J. Berstel, P. Chrétienne Eléments d'algorithmique, Masson, 2005
- T. Cormen, C. Leicerson, R. Rivalst Algorithmique, éditions Dunod, 2013

Matière : KARI6M09 - LANGAGE C POUR LA PROGRAMMATION SYSTEME (CPS) (C Language for System Programming)

Objectifs

Maîtrise du langage C, en particulier pour la programmation "bas niveau" (proche du système et du matériel).

Intended learning outcomes

Mastery of the C language, especially for "low level" programming (close to the system and the hardware).

Pré-requis

Niveau débutant en C (syntaxe de base, instructions conditionnelles et itératives, entrées/sorties conversationnelles, fonctions, tableaux).

Prerequisites

Beginner level in C (basic syntax, conditional and iterative instructions, conversational inputs / outputs, functions, tables).

Plan du cours

- Rappels sur les notions de base.
- Types de C (arithmétiques, caractères, chaînes de caractères) et leur codage.
- Tableaux et pointeurs (manipulation, gestion de la mémoire, passage en paramètre, arithmétique sur les adresses,...).
- Approfondissement sur les chaînes de caractères.
- Programmation modulaire, compilation séparée, Makefile; création de bibliothèques.
- Opérateurs bit à bit.
- nStructures et listes chaînées (création, destruction, manipulation).

Course content

- Survey of basic notions (conditional and iterative statements, I/O, functions, arrays).
- Types of C (arithmetic, characters, strings) and their encoding.
- Arrays and pointers (memory management, parameter passing, arithmetic on addresses,...).
- More on strings.
- Modular programming and compiling, Makefile; creation of libraries.
- Bitwise operators.

Structures and linked lists (creation, destruction, management).

Bibliographie

1. "Programmer en langage C", Claude Delannoy, Eyrolles.
2. "C: langage, bibliothèque, applications", Henri Garreta, InterEditions.
3. "Le langage C - Norme ANSI", B.Kernighan et D.Ritchie, Dunod.
4. "Managing projects with GNU Make", Robert Mecklenburg, O'Reilly.

Module : KARI6U04 - UE4: ARCHITECTURE-SYSTEME 2 (Architecture-System 2)

Matière : KARI6M10 - COMMUNICATIONS NUMERIQUES (CN) (Numerical Communications (DC))

Objectifs

Introduire la notion de communication numérique qui fait le lien entre le traitement du signal, l'électronique et les réseaux informatique. On retrouve pour cela les notions de signal, information binaire et paquets de données qui permettent d'établir une communication numérique.

Ce cours décrit les concepts, méthodes et techniques utilisées pour les communications numériques. Il se place toujours du point de vue de l'informatique des réseaux afin d'introduire les bases pour la compréhension des communications numériques nécessaires aux protocoles de niveau supérieur.

Suivant la référence au modèle OSI, on peut placer ce cours autour de la couche physique et de la partie liée au médium de la couche liaison.

Intended learning outcomes

The course introduces Digital Communications which link signal processing, electronics and computer networks. We study the concepts of signal, binary information and data packets that allow to establish a digital communication.

This course describes the concepts, methods and techniques used for digital communications. It always takes the point of view of network computing in order to introduce the basics for understanding the digital communications required for higher level protocols.

According to the reference to the OSI model, this course can be placed around the physical layer and the link-related part of the medium.

Pré-requis

Cursus scientifique, bases en informatique, techniques de base en programmation C pour les TP.

Prerequisites

Science curriculum, basic computer science, basic C programming techniques for practical work.

Plan du cours

Sont abordés, de manière plus ou moins approfondie suivant les besoins, les thèmes suivants:

- Représentation des signaux, canal physique, distorsion, bruit, filtrage, modulation
- Théorie de l'information, entropie, codage, compression, correction d'erreur
- Lien de communication, trame, contrôle d'accès au support, commutation de paquets, multiplexage, fiabilité, contrôle de flux

Le cours présente ces éléments avec un point de vue informatique et réseaux afin de mieux comprendre l'intérêt des différentes théories scientifiques sous-jacentes au fonctionnement des communications numériques.

Course content

The following domains will be covered:

- signal representation, physical channel, distorsion, noise, filter, modulation
- information theory, entropy, coding, compression, error correction
- communication link, frame, medium access control, packet switching, multiplexing, reliability, flow control

Lectures stays with a computer science and networking point of view in order to allow a better understanding of the underlying fundamental science beneath numerical communications.

Bibliographie

- Réseaux, A Tanenbaum, InterEditions
- Réseaux et Télécoms, C Servin, Dunod
- Les réseaux, G Pujolle, Eyrolles
- Transmissions et réseaux, Stéphane Lohier
- Réseaux informatiques, D Dromard
- Réseaux et communications sans fil, W. Stallings, Pearson
- MIT online lectures

Matière : KARI6M11 - RESEAUX (RES) (Networks (NS))

Objectifs

Cet enseignement est une introduction aux nombreuses problématiques liées aux transferts d'information entre ordinateurs.

Premières notions d'administration de réseau : montage, configuration système, observations, compréhension des problématiques résolues dans les protocoles d'Internet (Ethernet, IP, UDP, TCP)

Intended learning outcomes

The course is an introduction to the many issues related to information transfer between computers.

First notions of network administration: assembly, system configuration, observations, understanding of Internet protocols (Ethernet, IP, UDP, TCP).

Pré-requis
Notions élémentaires d'algorithmique
Prerequisites
Basic notions of algorithmic
Plan du cours
<ul style="list-style-type: none"> - Supports et codage physique - Mise en paquet de l'information - Accès multiple au support (protocole Ethernet) - Détection et correction d'erreurs - Adressage et routage (protocole IP) - Récupération d'erreur par re-émission (protocole TCP) - Contrôle de flux (protocole TCP)
Course content
<ul style="list-style-type: none"> - Supports and physical coding - Packaging of information - Multiple access to media support (Ethernet protocol) - Error detection and correction - Error Retrieval by resend - Flow control - Addressing and routing (IP protocol)
Bibliographie
<ol style="list-style-type: none"> 1. Analyse structurée des Réseaux - Des Applications de l'Internet aux infrastructures de télécommunication - James Kurose et Keith Ross - 2ème Edition - Pearson Education 2. Réseaux locaux et Internet (des protocoles à l'interconnexion) - Laurent Toutain - 2ème ou 3ème Edition - HERMES

Compétences
<p>Cet enseignement permet d'acquérir les premières compétences nécessaires à l'administration et l'utilisation intelligente des réseaux informatiques.</p> <p>A l'issue de ce cours les étudiants sont capables :</p> <ul style="list-style-type: none"> - d'assembler un réseau simple composé d'ordinateurs, de switches et de routeurs, - de configurer les ordinateurs et les routeurs pour le bon fonctionnement d'un réseau (configuration des interfaces réseaux, configuration via DHCP, tables de routage), - d'observer et d'analyser les entêtes protocolaires des paquets circulant sur le réseau, - d'appréhender facilement les problèmes liés au flux de paquets sur le réseau (partage de bande passante, tests de débits, estimation de latence...), - de programmer intelligemment une application client/serveur à l'aide de l'interface des sockets.

Matière : KARI6M12 - ARCHITECTURE LOGICIELLE ET MATERIELLE (ALM2) (Hardware and Software Architecture)
Objectifs
Le but de ce cours est d'acquérir une connaissance générale du fonctionnement bas niveau des ordinateurs. Ce cours s'intéresse à la fois aux aspects matériels (circuits logiques) des ordinateurs ainsi qu'aux aspects bas-niveau des systèmes d'exploitation. Les liens entre la conception du matériel et les services disponibles dans les systèmes d'exploitation sont particulièrement développés.
Intended learning outcomes
The purpose of this course is to gain a general knowledge of the low-level operation of computers. The course focuses on both the hardware aspects (logical circuits) of computers and the low-level aspects of operating systems. The links between hardware design and the services available in the operating systems are particularly well developed.
Pré-requis
Ce cours suppose la connaissance préalable de l'algèbre booléenne, des circuits logiques classiques (portes logiques, automates, ...), du langage assembleur ARM et quelques rudiments du langage C. Ces pré-requis sont principalement fournis dans l'UE ALM1.
Prerequisites
This course assumes prior knowledge of boolean algebra, classical logic circuits (logic gates, PLCs, ...), ARM assembler language and some C language basics. These prerequisites are mainly provided in part 1 of "Architecture Logicielle et Matérielle" which takes place during the previous semester.
Plan du cours
<ul style="list-style-type: none"> * Architecture interne d'un processeur (bus d'E/S, PC/PO, Entrées/Sorties, ...) * Interruptions (description générale, mise en œuvre dans les processeurs, influence sur l'architecture des processeurs, ...) * Introduction aux systèmes d'exploitation et aux processus (présentation générale, liens entre les fonctionnalités matériels et les services des systèmes d'exploitation, ...) * Gestion mémoire (segmentation, pagination, ...) * Caches (généralité, types de caches, implémentation matérielle, ...) * Techniques avancées dans les processeurs (pipeline, super-scalaire, exécution dans le désordre, ...) * Compilation, édition de lien, format ELF, relocation
Course content
<ul style="list-style-type: none"> * Internal processor architecture (I/O bus, OC/OP, Input/Output, ...) * Interruptions (description, implementation in processors, link with processors architecture, ...) * Introduction to operating systems and processes (description, links between hardware features and services of operating systems, ...) * Memory management (segmentation, pagination, ...) * Caches (description, kind of caches, hardware implementation, ...) * Advanced techniques in processors (pipeline, super-scalaire, out-of-order, ...) * Compilation, linking, ELF format, relocation

Bibliographie
<p>* processeurs et langage ARM</p> <p>** Processeurs ARM - Architecture et langage d'assemblage, Jacques Jorda, Dunod</p> <p>* Synthèse d'un processeur</p> <p>** Concevoir son microprocesseur, Jean-Christophe Buisson, Eyrolles / Technosup</p> <p>** Architecture et microarchitecture des processeurs, Bernard Goossens, Springer Verlag</p> <p>* Généraliste</p> <p>** Architecture de l'ordinateur, Andrew Tannebaum, Interéditions</p> <p>** Architecture des ordinateurs, Philippe Darche, Éd. Gaétan Morin</p> <p>** Logique booléenne : implémentations et technologies, Vuibert, Eyrolles/PUF</p> <p>** Fonctions booléennes, logiques combinatoires et séquentielle, Vuibert, Eyrolles/PUF</p> <p>** Interfaces et périphériques, Vuibert, Eyrolles/PUF</p> <p>** Représentation des nombres et codes, Vuibert, Eyrolles/PUF</p> <p>* Circuits combinatoires, automates</p> <p>** Architecture des machines et des systèmes d'exploitation, Alain Cazes & Joëlle Delacroix, Dunod</p> <p>* Portes logiques, circuits combinatoires, arithmétique binaire, circuits séquentiels et mémoires</p> <p>** Architecture de l'ordinateur, Robert Strandh et Irène Durand, Dunod sciences Sup</p> <p>** Computers 1 - Bit après bit, Jean Jacques Mercier, Eyrolles/ellipses</p> <p>** Séquences après séquences - Circuits asynchrones et synchrones, Jean Jacques Mercier, Eyrolles/ellipses</p> <p>*</p> <p>Autres</p> <p>** Du binaire au processeur, Emmanuel Mesnard, Eyrolles/Ellipses</p> <p>** De la diode au microprocesseur, Patrick Demirdjian, Technip Sciences et Technologies</p>

Module : KARI6U05 - UE5 : PROJETS ET ACTIVITES SPORTIVES (Projects and sport activities)

Matière : KARI6M13 - PROJET LOGICIEL NIVEAU SYSTEME (PLS) (Team Project in System Programming (SPS))

Objectifs

Découvrir les outils pratiques et intellectuels nécessaires à la gestion d'un projet informatique : développement à plusieurs sur un temps long.

Les connaissances sont mises en pratique dans le cadre du projet logiciel Applicatif (PLA) qui se déroule en parallèle.

Intended learning outcomes

A tour of the practical and intellectual tools needed to manage an IT project: development with a team over a long time.

The knowledge is put into practice during the application software project (PLA) which runs in parallel.

Pré-requis

/\ PLS devient Gestion de Projet Informatique (GPI)

/\ Le programme de PLS est passé en ALM2 avec des TP conséquents sur la programmation C bas niveau.

Savoir administrer sa machine linux, connaître le langage Java et l'environnement Eclipse

Prerequisites

Being able to install a linux machine, being at ease in Java and the Eclipse framework

Plan du cours

1. Développement (2 CM)

- Prototypage : rôle et limites
- Stratégie de tests et de développement
- Intégration continue (JUnit)
- Debugger

2. Pilotage de projet (3 CM)

- Système de gestion de version (Git) :
principe, commandes et bonnes pratiques
- Jalons et objectifs à long terme
- Interface et documentation
- Communication et répartition des tâches
- Analyse de l'impact écologique et énergétique

Course content

1. Development (2 CM)

- Prototyping: role and limits
- Test and development strategy
- Continuous integration (Unit)
- Debugger

2. Project management (3 CM)

- Version Management System (Git):
principle, commands and good practices
- Milestones and long-term goals

- Interface and documentation
- Communication and distribution of tasks
- Analysis of the ecological and energy impact

Matière : KARI6M14 - PROJET LOGICIEL NIVEAU APPLICATIF (PLA) (Software project: the application level (SPA))

Objectifs

Le projet met en jeu les compétences théoriques et pratiques acquises durant l'année 3. Il s'agit de développer un logiciel sur 3 semaines à plein temps par groupe de 5 ou 6 élèves.

Cet enseignement a pour but

- de renforcer les acquis
- illustrer les notions de gestion de projet et de génie logiciel.

Intended learning outcomes

This project involves the theoretical and practical skills acquired during year 3. It involves developing a software program for 3 weeks full-time in groups of 5 or 6 students.

This course aims at

- strengthening the abilities
- illustrating the notions of project management and software engineering.

Pré-requis

Avoir suivi les cours de l'année 3 en programmation impérative et objets, automates, analyse syntaxique.

Prerequisites

Having taken the courses of year 3 in imperative and object-oriented programming, automata and syntactic analysis is required

Plan du cours

Le cadre imposé est la réalisation d'un simulateur avec multiples objets en interaction. Les applications finales peuvent être très diverses selon ce qu'on simule : jeux, monde physique, protocoles, ...

Les sujets ont en commun les concepts suivants :

- le comportement de chaque objet actif est défini par un automate.
- l'analyse lexicale : lecture et interprétation d'un fichier de configuration (décor, comportement, équipe)
- la programmation par événement : couplage de l'interface graphique et des calculs (affichage, interaction, mise à jour)
- l'ordonnancement (scheduling) : simulation fluide de plusieurs éléments actifs simultanément sans utiliser de threads

Le projet est l'occasion d'une introduction au génie logiciel à travers

- l'utilisation d'un système de gestion de version (git)
- l'importance de l'architecture d'un logiciel
- le rôle des tests d'intégration
- la réalisation de documentations développeur / utilisateur
- la préparation d'une démo.

Le projet comporte 2 phases :

- 1ère semaine : réflexion, réalisation d'un prototype, présentation d'un projet, établissement d'un contrat
- 2e et 3ème semaines : développement, soutenance et démonstration, évaluation du respect du contrat

Course content

The imposed framework is the realization of a simulator with multiple interacting objects. The final applications can be very different depending on what is simulated: games, physical world, protocols, ...

The topics have in common the following concepts:

- the behavior of each active object is defined by an automaton
- lexical analysis: reading and interpreting a configuration file (scenery, behavior, team)
- Event programming: coupling of the graphical interface and calculations (display, interaction, update)
- scheduling: fluid simulation of several active elements simultaneously without using threads

The project is an opportunity for an introduction to software engineering through

- the use of a version management system (git)
- the importance of software architecture
- the role of integration tests
- the realization of developer / user documentation
- the preparation of a demo.

The project has 2 phases:

- 1st week: brainstorming, realization of a prototype, presentation of a project, establishment of a contract
- 2nd and 3rd weeks: development, defense and demonstration, evaluation with respect to the contract

Bibliographie

Exemples de sujet des années passées :

- jeu éducatif qui illustre les concepts de programmation (inspiré du serious game LigthBot)
- jeu de plateau où chaque élément réagit aux mêmes événements par des actions différentes
- simulation de compétition de robot dont le comportement est défini par un automate
- jeu classique avec nombre de joueurs arbitraires

site web du projet : <http://www-verimag.imag.fr/~perin/enseignement/RICM3/infpl2/>

Course literature
<p>Upon completion of this project the student has a first draft of experience.</p> <p>He faced some difficulties</p> <ul style="list-style-type: none"> - leading a project team - defending a proposal and leading it to its realization <p>He became aware of the importance</p> <ul style="list-style-type: none"> - to plan, delegate and communicate - to organize a code of average size - to reuse existing libraries

Compétences
<p>À l'issue de ce projet l'élève a une première expérience de projet.</p> <p>Il a été confronté aux difficultés</p> <ul style="list-style-type: none"> - de mener un projet en équipe - de défendre une proposition et de la mener à sa réalisation <p>Il a pris conscience de l'importance</p> <ul style="list-style-type: none"> - de planifier, déléguer et communiquer - d'organiser un code de taille moyenne - de réutiliser des bibliothèques existantes

Matière : KARI6M15 - MISE EN SITUATION VIA LE SPORT (Sport and Social Enhancement)

Objectifs

<p>L'objectif de ce cycle de sport en première année est double : progresser dans l'activité et développer ses propres compétences sociales.</p> <ul style="list-style-type: none"> - Apprendre à gérer de nouvelles situations, souvent complexes, à risques et stressantes. - Formaliser, verbaliser les difficultés rencontrées, travailler en équipe. - Mieux se connaître, se dépasser, résister à l'effort. - Anticiper, identifier, résoudre seul et en groupe les problèmes qui se présentent dans la pratique de l'activité. - Intégrer le sport dans un équilibre de vie, dans une stratégie de bien être, de santé et de sécurité. - Favoriser l'intégration, la solidarité.

Intended learning outcomes

<p>The goal of this first year sport cycle is twofold: to grow in activity and develop one's own social skills.</p> <ul style="list-style-type: none"> - Learn how to handle new situations, often complex, risky and stressful. - Formalize, verbalize the difficulties encountered, work in teams. - To know each other better, to go beyond, to resist to the effort. - Anticipate, identify, resolve alone and in groups the problems that arise in the practice of the activity. - Integrate sport in a balance of life, in a strategy of well-being, health and safety. - Promote integration, solidarity.
--

Pré-requis

<p>Aucun pré requis n'est nécessaire.</p> <p>Les activités support de l'enseignement seront certainement nouvelles pour la plupart des étudiants, le niveau technique dans l'activité n'est pas un critère discriminant.</p>
--

Prerequisites

<p>No prerequisites are necessary.</p> <p>Support activities for teaching will certainly be new for most students, the technical level in the activity is not a discriminating criterion.</p>

Plan du cours

<p>Au travers d'une activité sportive support de l'enseignement, l'étudiant identifie un problème qui le concerne dans sa pratique et propose des solutions pour parvenir à surmonter/résoudre ce problème. Ces solutions sont testées sur le terrain tout au long du cycle, puis validées, ou non.</p> <p>Ainsi, plus que la notion de niveau technique dans l'activité, c'est vraiment la capacité de l'étudiant à analyser sa pratique et son comportement qui nous intéresse ici.</p>

Course content

<p>Through a sporting medium of instruction, the student identifies a problem that concerns him in his practice and offers solutions for how to overcome / solve this problem. These solutions are field-tested throughout the cycle, then validated, or not.</p> <p>Thus, as the notion of technical level in the business, it's really the ability of students to analyze their practices and behavior of interest here.</p>
--