Bachelor in: Computer Science

Speciality: Computer Systems

The academic Bachelor in Computer Systems consists of three years of learning/training (six pedagogical semesters) in information technologies. The study program allows the student to acquire all the lessons that will lead him to build a theoretical and practical base of fundamental knowledge in computer science.

Field	Branch	Speciality
Mathematics and Computer Science	***Computer Science***	***Computer Systems***

Semester 1

Teaching unit	Subject	Credit	Coefficient	Course	TD	TP	HV
Fundamental Unit 1	Analysis 1	6	4	3h	3h		90h
	Algebra 1	5	2	1h30	1h30		45h
Fundamental Unit 2	Algorithmic and data structures 1	6	4	3h	1h30	3h	112h30
	Machine structure 1	5	3	1h30	1h30		45h
Methodological Unit	Scientific terminology and expression	2	1	1h30			22h30
	English 1	2	1	1h30			22h30
Discovery Unit	Physics 1	2	2	1h30	1h30		45h

Semester 2

Teaching unit	Subject	Credit	Coefficient	Course	TD	TP	HV
Fundamental Unit 1	Analysis 2	6	4	3h	1h30		67h30
	Algebra 2	4	2	1h30	1h30		45h
Fundamental Unit 2	Algorithmic and data structures 2	6	4	1h30	1h30	1h30	67h30
	Machine structure 2	4	2	1h30	1h30		45h
Methodological Unit	Introduction to probabilities and statistics	3	2	1h30	1h30		45h
	Information and communication techniques	2	1	1h30			22h30
	Mathematics programming tools	2	1	1h30		1h30	45h
Transversal Unit	Physics 2	2	2	1h30	1h30		45h

Semester 3

Teaching unit	Subject	Credit	Coefficient	Course	TD	TP	HV
Fundamental Unit 1	Computers architecture	5	3	1h30	1h30	1h30	67h30
	Algorithmic and data structures 3	6	3	3h	1h30	1h30	90h
Fundamental Unit 2	Information systems	5	3	1h30	1h30	1h30	67h30
	Graph theory	4	2	1h30	1h30		45h
Methodological Unit	Numerical methods	2	1	1h30		1h30	45h
	Mathematical logic	4	2	1h30	1h30		45h
Transversal Unit	English 2	2	1	1h30			22h30

Semester 4

Teaching unit	Subject	Credit	Coefficient	Course	TD	TP	HV
Fundamental Unit 1	Languages theory	5	2	1h30	1h30	1h30	67h30
	Exploitation systems 1	5	3	1h30	1h30	1h30	67h30
Fundamental Unit 2	Data bases	5	3	1h30	1h30	1h30	67h30
	Networks	5	3	1h30	1h30	1h30	67h30
Methodological Unit	Object-oriented programming	4	2	1h30		1h30	45h
	Web applications development	4	2	1h30		1h30	45h
Transversal Unit	English 3	2	1	1h30			22h30

Semester 5

Teaching unit	Subject	Credit	Coefficient	Course	TD	TP	HV
Fundamental Unit 1	Exploitation systems 2	5	3	1h30	1h30	1h30	67h30
	Compilation	5	3	1h30	1h30	1h30	67h30
Fundamental Unit 2	Software engineering	5	3	1h30	1h30	1h30	67h30
	Human Machine Interface	5	3	1h30	1h30	1h30	67h30
Methodological Unit	Linear programming	4	2	1h30	1h30		45h
	Probabilities and statistics	4	2	1h30	1h30		45h
Transversal Unit	Digital economy and strategic monitring	2	1		1h30		22h30

Semester 6

Teaching unit	Subject	Credit	Coefficient	Course	TD	TP	HV
Fundamental Unit 1	Mobile apps	5	3	1h30		1h30	45h
	Information security	5	3	1h30	1h30		
Fundamental Unit 2	Artificial intelligence	5	3	1h30		1h30	45h
	Semi-Structured data	5	3	1h30		1h30	45h
Methodological Unit	Project	6	3				150h
	Scientific redaction	2	1		1h30		22h30
Transversal Unit	Create and develop a startup	2	1	1h30			22h30

Detailled program

Semestre 3 : Computer science
Fondamental Teaching Unit (FTU)
Subject : Computers architecture
Crédits : 5
Coefficient : 3

Teaching goals: The aim of this subject is to clarify the principle of the computer functioning with a detailed presentation of the computer architecture.

Prerequisites :

Subject Content

Chapter 1 :

- Introduction and Basics
- Von Neumann and Harvard architecture

Chapter 02: Principals components of a computer

- Global schema architecture
- ALU Arithmetic and Logic Unit
- Bus
- Registers
- Internal memory: RAM memory (SRAM et DRAM), ROM, access time, latency,...
- Cache memory: Utility and principle, cache memory management algorithms (Basics)
- Memory hierarchy.

Chapter 03: The processor instruction set

- High-level programming language, Assembly language and machine language
- Instruction set (Arithmetic, Logic, comparison, Load, Store, Transfer and Jump)
- Compilation and assembly principles (Basics)
- The control unit
- Instruction execution phases (Fetch, Decode, Address Calculations, execution)
- pipelining
- Clock and Sequencer

Chapter 4: The processor

- The processor function, CPI calculation (Cycle per Instruction), CISC and RISC architecture.
- The MIPS R3000 processor
- Extern architecture of the MIPS R3000 processor
- Internal architecture of the MIPS R3000 processor
- MIPS R3000 Instruction set.
- MIPS R3000 programming.

Chapitre 5 : Spécial instructions

Interrupts, Inputs/Outputs and system instructions (MIPS R3000)

Evaluation mode: Exam (60%), CC (40%)

References

- Alain Cazes , Joëlle Delacroix, Architecture des machines et des systèmes informatiques 4 ème édition, *Collection : Informatique, Dunod, 2011.*
- Andrew S. Tanenbaum, Todd Austin Structured Computer Organization, Pearson, 2012.
- Paolo Zanella, Yves Ligier, Emmanuel Lazard, Architecture et technologie des ordinateurs : Cours et exercices - *Collection : Sciences Sup, Dunod*, 5ème édition, 2013.
- Liens vers le microprocesseur MIPS R3000
  ftp://132.227.86.9/pub/mips/mips.asm.pdf
  ftp://asim.lip6.fr/pub/mips/mips.externe.pdf
  ftp://asim.lip6.fr/pub/mips/mips.interne.pdf

Semester 3 : Computer science

Fundamental Teaching Unit (FTU)

Subject: Algorithmics and data structures 3

Credits: 6

Coefficient: 3

Teaching goals: This module will allow students to learn on the one hand the development of certain basic algorithms in computer science, on the other hand, they will learn to manipulate more developed data structures.

Recommended prior knowledge: basic algorithms

Subject content:

Reminder

Chapter 1: Algorithmic Complexity

1. Introduction to complexity
2. Complexity calculation

Chapter 2: Sorting Algorithms

1. Presentation
2. Bubble Sort
3. Sort by selection
4. Insertion sort
5. Merge Sort
6. Quick Sort Chapter

Chapter 3: Trees

1. Introduction
2. Definitions
3. Binary Tree
    - **a.** Definition Transition from an n-ary tree to a binary tree
    - **b.** Chained representation of a binary tree
    - **c.** Path of a binary tree
        - **i.** Prefixed route (preorder or RGD)
        - **ii.** Infixed course (projective, symmetric or even GRD)
        - **iii.** Postfixed route (terminal order or GDR)
        - **iv.** Particular binary trees
        - **v.** Complete Binary Tree
        - **vi.** Search Binary Tree

Chapter 4: Graphs

1. Definition
2. Representation of graphs
3. Itinerary of the graphs

N.B : Lab in C language

Evaluation mode : Exam (60%), CC (40%)

References

- Thomas H. Cormen, Algorithmes Notions de base *Collection : Sciences Sup, Dunod,* 2013.
- Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest Algorithmique - 3ème édition - Cours avec

957 exercices et 158 problèmes Broché, Dunod, 2010.

- Rémy Malgouyres, Rita Zrour et Fabien Feschet. Initiation à l’algorithmique et à la programmation en C : *cours avec 129 exercices corrigés*. 2ième Edition. Dunod, Paris, 2011. ISBN : 978-2-10-055703-5.
- Damien Berthet et Vincent Labatut. *Algorithmique & programmation en langage C - vol.1 : Supports de cours*. Licence. Algorithmique et Programmation, Istanbul, Turquie. 2014, pp.232.
- Damien Berthet et Vincent Labatut. *Algorithmique & programmation en langage C - vol.2 : Sujets de travaux pratiques*. Licence. Algorithmique et Programmation, Istanbul, Turquie. 2014, pp.258. <cel-01176120>
- Damien Berthet et Vincent Labatut. *Algorithmique & programmation en langage C - vol.3 : Corrigés de travaux pratiques*. Licence. Algorithmique et Programmation, Istanbul, Turquie. 2014, pp.217. <cel-01176121>
- Claude Delannoy. *Apprendre à programmer en Turbo C*. Chihab- EYROLLES, 1994.

Semester 3 : Computer science

Fundamental Teaching Unit (FTU)

Subject: Information systems
Credit : 5
Coefficient : 3

Teaching objectives:

1. Understand what an enterprise information system is;
2. To understandthe different constituent dimensions of an IS:
   - Technical size
   - Organizational dimension
   - Managerial dimension
3. Understand the different elements of an IS:
   - piloting system
   - system decision-making
   - system operational
4. To understand the joint of WHETHER with there strategy corporate (governance of WHETHER - management IS projects)

Knowledge prerequisites recommended: Algorithmic,

Subject Content
Chapter 1 : Generality

- Definitions And characterizations of the company (Aspects functional and structural),
- Systemic approach of organizations:
  1. Overview of the three systems (Decision system, classification of decisions: by level and by method, a techniqueprogrammable decision tables, decision tables),
  2. The information system (Aspects functional and structural aspects: notion of station, workstation, flow, documents),The diagram of flow.

Chapter 2: Techniques of representation of information

Notion of information, Forms and manipulation of information, Study of information:Class and class realization, class description, … Diagram and codification of information

Chapter 3: Seizure And control of information

Different guys of control of information

Chapter 4: Methodology of development of one WHETHER: CHERRY

- Process of development of one WHETHER
- Level of abstraction of model of data And treatments
- Methodology CHERRY
- Concepts for static modeling (Notion of entity and association, a model data concept: the MERISE MCD. Concepts for dynamic modeling: MCTof MERIS.

Evaluation mode: Exam (60%), CC (40%)

References

- Coord . p. Vidal, p. Planeix , Systems of information organizational, 2005.
- Coord . ML. Caron- Fasan & NOT. Lesca , present And future of the information system, 2003, PUGS. p.
- Kalika Mr. & ali , THE e-management. What transformations For the company ? , 2003, Editions Connections .
- JLLemoigne , There theory of general system. PUF-
- v. Bertalanfy , Theory general of the systems. Dunod .
- X. castellani, Method general analysis of one application computer science. Mason, 1975.
- Tardieu And para. , " there method sweet cherry : principles And tools », ed. organization, 1983.-
- Tardieu And para. , " there sweet cherry method : Steps and practical » ed. organization, 1985.-
- Stool , " the other coast by Merise », ed. organization, 1986.-
- J. p. Matheron , " To understand Merise », 1990

Semester 3 : Computer science

Fundamental Teaching Unit (FTU)

Subject: Graph Theory

Credits: 4

Coefficient: 2

Teaching goals: Graph theory has become a theoretical and practical foundation essential in the process of modeling of certain problems in several domains. The contribution of graphs in solving problems lies in the simplicity of its representation graphic, the similarity with distributed aspects and the notions of traversal and path searches. The objective of this course is to present to the student on the one hand a solution modeling in the form of a graph, on the other hand this course will contain a set of techniques allowing the student to solve his problems through algorithms like minimum path search, maximum flow etc.

Subject content

Chapter I. Basic definitions

1. "Intuitive" definition of a graph
2. Mathematical definition of a graph
3. Order, orientation and multiplicity
    - 3.1. Order
    - 3.2. Orientation
    - 3.3. Multiplicity
4. Relationships between the elements of a graph
    - 4.1 Relationships between vertices
    - 4.2 Relationships between arcs and vertices
    - 4.3 Graph qualifiers
5. Matrices associated with a graph
    - 5.1 Vertex-arc incidence matrix
    - 5.2 Adjacency matrix or vertex-vertex incidence matrix
    - 5.3 Condensed form of sparse matrices
6. Vocabulary related to connected graph
    - 6.1 Chain, path, length
    - 6.2 Connected graph
    - 6.3 Cycle and circuit
    - 6.4 Cocycle and Cocircuit.

Chapter II. Cycles

1. Cyclomatic and cocyclomatic numbers
    - 1.1 Decomposition of cycles and cocycles into elementary sums
    - 1.2 Arc Colouring Lemma (Minty 1960)
    - 1.3 Cycle basis and cocycle basis
2. Planarity
    - 2.1 Planar Graph
    - 2.2 Euler's formula
    - 2.3 Kuratowski's theorem (1930)
    - 2.4 Dual graph
3. Tree, forest and arborescence
    - 3.1 Definitions
    - 3.2 Properties
    - 3.3 Maximum (or spanning) tree

Chapter III. Flow problems

1. Definitions
2. Search for a maximum flow in a transport network

2.1 Definition

2.2 Ford-Fulkerson theorem

2.3 Ford-Fulkerson algorithm

3. Search for a compatible flow

Chapter IV. Pathway problems

1. Search for connected components

1.1 Presentation of objectives

1.2 Trémeaux-Tarjan algorithm

2. Finding the shortest path

2.1 Presentation of the conditions

2.2 Moore-Dijkstra algorithm

3. Search for a spanning tree of maximum weight in a graph

3.1 Presentation of objectives

3.2 Kruskal Algorithm 1956

Chapter V. Hamiltonian and Eulerian Problems

1. Hamiltonian Problem

1.1 Definitions

1.2 Necessary condition for the existence of a Hamiltonian cycle

1.3 Sufficient condition for the existence of a Hamiltonian circuit

1.4 Sufficient condition for the existence of a Hamiltonian cycle

2. Eulerian Problem

2.1 Definitions

2.2 Necessary and sufficient condition for the existence of an Eulerian chain

2.3 Local algorithm to construct an Eulerian cycle

2.4. Relation between Eulerian and Hamiltonian problem

Chapter VI. Colouring

1. Definitions
2. Vertex colouring
3. Arc colouring
4. Proposals
5. The "4 colours" theorem
6. Perfect graph

Evaluation mode: Exam (60%) , CC (40%)

References

- Claude Berge, Graphes et hypergraphes, Bordas 1973, (300 pages).
- Nguyen Huy Xuong, Mathématiques discrètes et informatique, Masson, 1997
- Aimé Sache, La théorie des graphes, Que-Sais-Je ?, 1974 ; réédition prévue en 2004 chez Cassini.
- M. Kaufmann, Des points des flèches, la théorie des graphes, Dunod, Sciencespoche, épuisé.
- Alan Gibbons, Algorithmic graph theory, Cambridge University Press, 1985
- Reinhard Diestel, Graph Theory, Second Edition, Springer-Verlag, 2000.
- Bojan Mohar, Carsten Thomassen, Graphs on surfaces, John Hopkins University Press, Baltimore, 2001.

Semester 3 : Computer science
Methodological Teaching Unit (MTU)
Subject: Numerical Methods
Credits: 4
Coefficient: 2
Teaching aim: this subject will allow students to invest in the field of numerical methods necessary to the resolution of problems
Recommended prerequisites: Basic mathematics
Content of the subject:

Chapter 1: Generalities of numerical analysis and scientific computing.
1.1 Motivations.
1.2 Floating point arithmetic and rounding errors
1.2.1 Representation of numbers in a machine
1.2.2 Rounding errors
1.3 Stability and error analysis of numerical methods and conditioning of a problem

Chapter 2: Direct Methods for Solving Linear Systems
2.1 Remarks on the solution of triangular systems
2.2 Gauss elimination method
2.3 Matrix interpretation of Gauss elimination: the LU factorization

Chapter 3: Iterative methods for solving linear systems
3.1 Generalities
3.2 Jacobi and over-relaxation methods
3.3 Gauss-Seidel and successive over-relaxation methods
3.4 Remarks on the implementation of iterative methods
3.5 Convergence of the Jacobi and Gauss-Seidel methods

Chapter 4: Computation of eigenvalues and eigenvectors
4.1 Localization of eigenvalues
4.2 Power method

Chapter 5: Matrix Analysis
5.1 Vector Spaces
5.2 Matrices
5.2.1 Operations on matrices
5.2.2 Links between Linear Applications and Matrices
5.2.3 Inverse of a matrix
5.2.4 Trace and determinant of a matrix
5.2.5 Eigenvalues and Eigenvectors
5.2.6 Similar matrices
5.2.7 Some particular matrices
5.3 Norms and Scalar Products
5.3.1 Definitions
5.3.2 Scalar Products and Vector Norms
5.3.3 Norms of matrices....

Evaluation method: Exam (60%), CC (40%).

References:

-M. Schatzman Analyse numérique : une approche mathématique, Dunod 2004.
-P.G. Ciarlet, Introduction à l’analyse matricielle et à l’optimisation, Masson 1990.
-J. Demmel, Applied Numerical Linear Analysis, SIAM 1997 ;
-C. D. Meyer, Matrix Analysis and Applied Linear Algebra, SIAM 2000 ;
-P. Lascaux et J. Théodor, Analyse numérique matricielle appliquée à l’art de l’ingénieur, 2 tomes, Masson 1988.
-G. H. Golub, C. F. van Loan, Matrix Computations, The Johns Hopkins University Press, 1989.

Semester 3 : Computer science
Methodological Teaching Unit (MTU)
Subject : Mathematical logic
Credit : 4
Coeffecient : 2
Teaching goals : Human reasonning formalization.
Prerequisites : Basic knowledge in Mathematics, Boolean Algebra
Subject content :

Chapter 1 : Introduction

1.1 Objective of matematical logic
1.2 Syntax and Semantics

Chapter 2 : Logic of propositions

2.1 Syntax
- Proposition
- Logical connectors
- Variables and propositonnal formulas
- Substituion in a formula
- Logical formulas and trees

2.2 Semantics
- Interpretation
- Truth table
- Tautology and Antilogy
- Semantic equivalence
- Conjunctive and disjunctive normal forms
- Satsfiability and Validity

2.3 Resolution
- Refutation
- Clausal form
- Propositional resolution rule
- Propositional resolution method

Chapter 3 : Logic of predicates

3.1 Syntax
- Terms
- Predicates
- Quantifier
- Formulas
  - Scope of an identifier
  - Free variables and bound variables

3.2 Semantics
- Structure
- formula satisfaction

Evaluation mode : Exam (60%), CC (40%)

References

- S.C. Kleene. Logique mathématique. Collection U, Armand Colin, Paris 1971.
- J.L. Krivine. Elements of Mathematical Logic. North-Holland Publishing Company Amsterdam, 1967.
- R. Cori. Logique mathématique. Tome 1 : Calcul propositionnel, Algèbre de Boole, calcul des prédicats. Dunod, 2003.

Semester 3 : Computer science
Transversal Teaching Unit (TTU)
Subject : Foreign language 2
Credit : 2
Coeffecient : 1
Teaching goals : Deepening and use of the English language in the manipulation of documents.
Recommended prerequisite knowledge: Good knowledge of English

Subject content:

- Didactic activities.
- Understanding of documents written in English.
- Reformulations.
- Written production.
- Translation exercises: French – English and English – French.
- Essays on writing small technical reports.

Evaluation mode: Exam (60%), CC (40%)

References

Murphy. English Grammar in Use. Cambridge University Press. 3rd edition, 2004
M. Mc Carthy et F. O'Dell, English vocabulary in use, Cambridge University press, 1994
L. Rozakis, English grammar for the utterly confused, Mc Graw-Hill, 1st edition, 2003
Oxford Progressive English books.

Semester 4 : Computer science
Fundamental Teaching Unit (MTU)
Subject : Language theory
Credit : 5
Coeffecient : 2
Teaching goals : understand the theory and tools of language theory
Recommended prerequisite knowledge: Basic math and computer skills

Subject content:
Chapter 1: Introduction and Objectives
Chapter 2: Alphabets, Words, Languages
Chapter 3: Grammars

1. Definitions
2. Derivation and generated language
3. Bypass shaft
4. Chomsky hierarchy

Chapter 4: Finite state automata (FSA)

1. Deterministic FSAs
2. Representation of an automaton
3. Equivalent and complete automata
4. Non-deterministic FSAs (determination)
5. Automata and regular languages (transformations and properties)

Chapter 5: Regular Expressions

1. Definitions
2. Kleene's theorem
3. Lemma of the star

Chapter 6: Minimizing an FSA
Chapter 7: Algebraic Languages

1. Properties of a regular grammar
2. Transformation of a regular grammar
3. Reduced grammar
4. Proper grammar
5. Elimination of left recursivities
6. Normal forms

Chapter 8: Battery-Powered Automata

1. Definition
2. Configuration, transition and calculation
3. Acceptance criteria
4. Deterministic battery-powered automata

Chapter 9: Turing Machine

1. Definition
2. Configuration, transition and calculation
3. Acceptance

Evaluation: Exam (60%), CC(40%)
References

1.P.Wolper. Introduction to computability. 2006, Dunod.
2. P. Seebold. Automaton theory. 2009, Vuibert.
3. J.M. Autebert Theory of languages and automata. 1994, Mason.
4. J. Hopcroft, J. Ullman. Introduction to Automata Theory, Languages and Compilation 1979, Addison- Wesley

Semester 4 : Computer science
Fundamental Teaching Unit (FTU1)
Subject: Operating System 1
Credits: 5
Coeffficient: 3
Teaching objective: to introduce the basic notions of operating systems in the management of machine resources: processor and central memory, then to present the mechanisms and techniques used for these purposes.
Recommended prior knowledge: algorithms and data structures, machine structure.

Subject content

Chapter 1: Introduction

- concept of operating systems
- Functions and roles.
- Example of operating systems: Windows – Unix- Android …

Chapter 2: Processor Management

- Definitions
  - Program concept.
  - Process concept.
  - Thread concept.
  - Notion of resource
  - Concept of work (Job)
- Different states of a process
- Process hierarchies.
- Relations between processes (competition, cooperation and synchronization).

-Process scheduling techniques:

-criteria: fairness, efficiency, response time, execution time, yield, etc.

-Scheduling algorithms (among the most used)

- Tourniquet (Round Robin RR).
- First-in, first-served or FCFS (First Come First- Served) algorithm.
- Shortest Job First (SJF) algorithm .
- Shortest Remaining Time Algorithm or SRT (Shortest Remaining Time).
- Algorithm with priority.

Chapter 3: Memory Management

- Goals.
- Monoprogramming.
- Multiprogramming:

a) Multiple contiguous partitions.

- Fixed contiguous partitions.
- Dynamic contiguous partitions:
  1- First Fit Strategy
  2- Best Fist Strategy
  3- Worst Fit Strategy .

a. Siamese contiguous partitions (Buddy system)
b. Re-allocation and protection
c. Back and forth (Swap)
d. Fragmentation and Compaction

- Multiprogramming and multiple non-contiguous partitions

1. Paging
2. Segmentation
3. Paginated segmentation.

- virtual memory
  - Virtual memory concept.
  - Overlays (overlay segments)
  - Pagination on demand
  - Some page replacement algorithms:

-Optimal algorithm
-Random Replacement
- chronological order of loading (FIFO) (with remarks on the anomaly of Belady)
- Chronological order of use (LRU: Least Recently Used).
-Frequency of use (LRU: Least Recently Used)
- Second chance algorithm.

Chapter 4: The Unix system.

Directed and practical work

The tutorials will focus on algorithm proposals around the different chapters. These algorithms will be developed in TP using the language in C language under Unix
The Unix system will be the focus of the first practical sessions .

Evaluation method : Exam (60%), CC (40%)

References

- Tanenbaum , Modern operating systems, third edition, Pearson, 2014
- ▪ Tanenbaum , Operating systems: courses and corrected exercises, Dunod , collection: sciences sup, 2004.
- Michel Divay , Unix, Linux and operating systems: course and corrected exercises, Dunod , collection : Sciences sup, 2004.
- Crocus, Computer Operating Systems, Dunod , 1993 .
- Sacha Krakowiak , Principles of Computer Operating Systems, Dunod , 1993

Semester 4: Computer science
Fundamental Teaching Unit (FTU)
Subject: Databases
Credits: 5
Coefficient: 3

Teaching goals: this course should allow the student to identify the interest of structuring and manipulating data in tabular form. Through the relational model and the underlying relational algebra oriented more towards the practical aspect. The student should understand the importance of structuring data, the concept of independence of data and processing, as well as the integrity and consistency of data.

Recommended prerequisites knowledge : Students are expected to understand what files are (Text, binary or typed) and to have created them with the previously studied languages.

Subject content

Chapter 1: Introduction to databases

1. Notions of files (interests and limits)
2. Database definition
3. Database Management System Definition
4. Types of data models (semantic, entity-association, hierarchical, network, relational)

Chapter 2: Relational model

1. Definition of the relational model
2. Basic Concepts (Attribute, Tuple, Domain, Relationship)
3. Relationship Diagram
4. Standardization To.
   a. Relationship Key and Functional Dependency (Primary Key and Foreign Key)
   b. Integrity Constraints vs.
   c. Normal forms (1FN, 2FN, 3FN, Boyce-Codd FN)
   d. Database schema
5. Logical relational model (SQL)
   a. Table, column, and row
   b. Description of SQL (Structured Query Language)
   c. Data definitions
      i. Create table (CREATE)
      j. Schema modification (ALTER, DROP)
   d. Data manipulation (INSERT, UPDATE, DELETE)

Chapter 3: Relational Algebra

1. Definition
2. Unary operations and operators
   a. Selection
   b. Projection vs. Translation in SQL
      i. Simple queries (SELECT-FROM)
      ii. Column selection (WHERE clause)
      iii. Sorting results (ORDER BY)
3. Set operations and operators
   a. Union
   b. Intersection vs. Difference
   c. Cartesian product
   d. Join (Theta, natural, equijoin, outer)
   e. Division g. Translation in SQL

i. Union, intersection and difference operators
ii. Cartesian product (no join)
iii. Joining tables (join condition)
iv. Aggregate function
v. GROUP BY … HAVING clause

Evaluation method: Exam (60%), CC (40%).
References:

- Bases de données. Georges Gardarin. 5ème édition 2003
- SQL Les fondamentaux du langage. Eric Godoc et Anne-Christine Bisson. Edition Eni. 2017
- Bases de données : concepts, utilisation et développement. Jean-Luc Hainaut. Édition DUNOD. 2015

Semester 4: Computer science
Fundamental Teaching Unit (FTU)
Subject: Networks
Credits: 5
Coefficient: 3

Teaching goals: This subject aims to give students the notions essential for a good understanding of networks. They must be able to explain what a network, what it is made of, how computers can communicate with each other, describe the different types of media, the different types of topologies as well as a detailed study of the different layers of the Internet model.

- Make the student able to understand the operation, to plan the installation and to use a network of computers.
- Familiarize the student with the various implementation layers of a computer network.
- Introduce the student to the main communication and message routing protocols.
- Familiarize the student with the main components of a computer network.
- Make the student able to use the basic services of a network within a program.

Recommended Prerequisite knowledge: Machine structure, components and systems

Subject Content:
Chapter 1: Introduction to Networks

1. Use of networks
2. Physical characteristics
3. Network topologies
4. Reference models (OSI, TCP/IP)
5. Types of gateways

Chapter 2: Physical Layer

1. Network Terminology
2. Signals, Decomposition, Noise
3. Guided and non-guided transmission supports
4. Digital transmission: Conversion from analog to digital
5. Digital transmission: Conversion from digital to digital
6. Sampling
7. Analog transmission: Conversion from digital to analog
8. Analog transmission: Conversion from analog to analog
9. Multiplexer and Concentrator

Chapter 3: Data Link Layer

1. Addressing
2. Flow control
3. 802.3 standard and Ethernet format
4. Error control
5. Multiple access control
6. Circuit switching

Chapter 4: Network Layer

1. IP addressing, classes, concept of subnets
2. IP protocol: IPV4, IPV6
3. Packet fragmentation
4. Packet switching
5. Routing: centralized techniques, distributed techniques
6. Static routing and dynamic routing
7. Hierarchical and external routing

Chapter 4: Transport Layer

1. Concept of transport address
2. UDP and TCP protocols
3. Quality of service
4. Congestion control

Chapter 5: Application Layer

1. SMTP protocol
2. HTTP protocol
3. FTP protocol
4. DHCP protocol
5. DNS protocol

Practical work

1. Lab 1: Basic network configuration
2. Lab 2: Network programming (Socket)
3. Lab 3: Routing
4. Lab 4: Protocol analyzer

Evaluation method : Exam (60%), CC (40%)

References

- Forouzan, Behrouz A., and S. C. Fegan. "Data communication and computer networks." (2007).
- Tanenbaum, Andrew S. "Computer networks, 4-th edition." ed: Prentice Hall (2003).

Semester 4 : Computer science
Methodological Teaching Unit (MTU)
Subject: Object Oriented Programming
Credits: 4
Coefficient: 2

Teaching goals: The basic objective of this course is to introduce the basic concepts of object-oriented programming (OOP) by practice using the Java language. Each chapter includes concepts that are translated at the end into java , so that the student can translate the theoretical concepts acquired into practice. At the end of the semester, the student is expected to have acquired the following skills:

1. The essence of object programming and its transformation into the Java language
2. To acquire an intuitive reasoning to give a solution to a simple problem according to the object-oriented
3. Write a program in Java that is functional
4. The essence and importance of OO reasoning and OOP

Prerequisite knowledge: C language

Subject content :
Chapter 1. Basics of OOP

1. Introduction
2. Fundamental concepts of OOP
    - a. Short history of OOP
    - b. Procedural programming vs. object-based programming
    - c. Code reuse
    - d. Introduction to modularity
3. Objects and classes
    - a. Notions of object
    - b. Notions of class
    - c. Attributes
    - d. Notion of message
    - e. Problem solving by message exchange
4. Introduction to Java
    - a. Types and control structures in Java
    - b. Classes and instantiation
    - c. Methods
    - d. References and parameter passing
    - e. Inputs /Outputs
    - f. Default constructor and other constructors
    - g. Destructors

Chapter 2: Encapsulation

1. Levels of visibility
2. Encapsulation
    - a. Encapsulation of data (attributes)
    - b. Code encapsulation (Messages)
3. Encapsulation in Java
    - a. Access control (public, private)
    - b. Accessors (get and set)
    - c. Access to the instance (this)
    - d. Class variables and methods (static)

Chapter 3: inheritance

1. Subclasses and inheritance
2. Simple inheritance, multiple inheritances
3. Class hierarchy
4. Polymorphism
5. Inheritance and polymorphism in Java
    - a. Simple inheritance (extends)
    - b. Encapsulation in heritage
        - i. Member protection (protected)
        - ii. Class constructors (this(), super())
        - iii. ‘Object’ class
        - iv. Implicit and explicit transtyping
        - v. Limitation of heritage (final)
        - c. Polymorphism
            - i. Method overloading
            - ii. Method redefinition
    - d. Abstract classes (use and importance)
    - e. Interfaces (use and importance)

Evaluation method: Exam (60%) , CC (40%)

References

1. Apprendre la Programmation Orientée Objet avec le langage Java. Luc Gervais. Eni. 2ème édition.
2. https://openclassrooms.com/courses/apprenez-a-programmer-en-java
3. Java 8 - Apprendre la Programmation Orientée Objet et maîtrisez le langage. Thierry GROUSSARD Luc GERVAIS. Edition ENI. 2015.
4. La programmation objet en Java. Michel Divay. Edition DUNOD. 2006.

Semester 4 : Computer science
Methodological Teaching Unit (MTU)
Subject: Web Applications Development
Credits: 4
Coefficient: 2
Educational Goals: The ultimate goal is to learn how to implement a web application.

Recommended prior knowledge: Fundamentals of algorithms and programming.
Basics of Internet and Networks.

Content:
Chapter 1: Introduction to World Wide Web

1. Definition and history
2. Client/Server architecture
3. HTTP protocol.

Chapter 2: Web Programming Languages

1. Overview: static page, dynamic page and web applications
2. Markup languages: definition and history
3. HTML
   3.1. What is HTML?
   3.2. HTML runtime environment
   3.3. Basic HTML
      3.3.1.Structure of HTML documents (header, body, links ...)
      3.3.2.Tables, Frames, Forms
      3.3.3.HTML 5.0
      3.3.4.Style Sheets (CSS 3)
      3.3.5.JavaScript
      3.3.6.Controlling HTML Forms in JavaScript
4. XML
   4.1. Structure of an XML document
   4.2. DTD (Document Type Definition)
   4.3. XML Schema
   4.4. XSLT

Chapter 3: Server-side programming language (PHP)

1. Introduction
2. Basic syntax
   2.1. Escaping from HTML
   2.2. Instruction separation
   2.3. Comments
3. Types, variables and operators
4. Control structures
5. Classes and objects
6. Features
   6.1. Error handling
   6.2. Managing file uploads
   6.3. Working with Remote Files
   6.4. Connection management
   6.5. Persistent Connections to Databases.
   6.6. Session management

    6.7. Business web applications in PHP

Chapter 4: Web Services: Basics

1. Introduction
2. Service Oriented Architecture (SOA)
3. Web Services Features
    3.1. Definition of web services
    3.2. Web Services Architecture
4. Web Services Standards
    4.1. SOAP
    4.2. WSDL
    4.3. UDDI
5. Web services development platforms
    5.1. Web services development (Server-side)
    5.2. Web services development (Client-side)
6. Platform .NET and Java.
    6.1. JSP
    6.2. ASP

Chapter 5: Case Study: Developing a Web Service (server-side then client-side)

Evaluation method: Exam (60%), continuous assessment (40%)

References:

- Web development courses. From: https://openclassrooms.com/courses.
- Jean Engels & Olivier Salvatori, "PHP 5: cours et exercices". Eyrolles editions, 2005
- Mathieu Lacroix, "Introduction Web: Cours". Université paris 13, 2013.
- Société Digimind. "Le Web 2.0 pour la veille et la recherche d'information, Exploitez les ressources du web social", Digimind, juin 2007.

Semester 4: Computer science
Transversal Teaching Unit (TTU)
Subject: Foreign language 3
Credits: 2
Coefficient: 1

Teaching goals: Techniques of written and oral expression in English: presentation, presentation, communication in groups. This course should leave as much freedom as possible to the students to be able to express themselves in English around a determined theme. Each group of students prepares a presentation that they will present in front of their classmates in English

Recommended Prerequisite knowledge: English of L1 level and Semeter 3.

Subject Content:

- Presentations
- Defense
- Communication in groups

Evaluation mode : Exam (100%)

Detailled program

Computer systems

Semester 5: Computer systems
Fundamental Teaching Unit (FTU1)
Subject: Operating system 2
Credits: 5
Coefficient: 3

Teaching goals: A thorough study of the Unix system is recommended during TD classes and practical TP labs. Threads programming and mutual exclusion are conducted with Clanguage under Unix. The producer/consumer, reader/writers and philosophers models with several variants will be studied theoritically and in the TD classes and implemented with C language in the TP labs.
Recommended Prerequisite knowledge: Operating system 1.

Subject Content:
Chapter 1 :

1. Recalls on the concept of OS.
2. Notions of program, process, thread and shared resource.

Chapter 1 : Process of synchronization

1. Problem of concurrent access to resources and critical section (mutual exclusion problem)
2. Synchronization tools:
   - Events, Locks
   - Semaphores
   - Monitors
   - Critical regions.
   - Path expressions

Chapter 3: Interprocess Communication

- Sharing of variables (models: producer/consumer, readers/writers)
- Mailboxes
- Message exchange (client/server model)

Chapter 4: Deadlock

- Templates
- Prevention
- Avoidance
- Detection/ Recovery

Evaluation method: Exam (60%) , CC (40%)

References:

1. Tanenbaum, Modern operating systems, third edition, Pearson, 2014
2. A. Tanenbaum, Systèmes d’exploitation, Dunod, 1994 .
3. Michel Divay, Unix, Linux et les systèmes d'exploitation : cours et exercices corrigés , 2004.
4. Crocus, Systèmes d'exploitation des ordinateurs, 1993.
5. Sacha Krakowiak, Principes des systèmes d'exploitation des ordinateurs, Dunod, 1993

Semester 5: Computer systems
Fundamental Teaching Unit (FTU)
Subject: Human-Machine Interface
Credits: 5
Coefficient: 3

Teaching goals: enable students to acquire skills in making visual graphic interfaces respecting the ergonomic criteria and design standards of the interactive and user-friendly interfaces.
- Knowledge of ergonomic rules
- Know what is interface development method
- Coupling with the method of development by objects
- Implementation of these methods in a project

Recommended Prerequisite knowledge: Algorithmics and data structures, Software engineering.

Subject Content:

Chapter 1: Notion of interaction

1. Definitions: Interaction, Interactivity, ...
2. Causes of rejection of certain applications.
3. Stakes: source of savings, change of workstation, consequences of a neglected interface neglected
4. Difficulties: the variety of users, difficult manufacturing, link between designer and developer
5. Definition of an HMI.
6. History of HMIs.

Chapter 2: Methodology for constructing an HMI

1. Classic Methodology.
2. Identification step: identification of functional areas, definition of the user model (notion of user profile), definition of the task model (types of tasks) technical environment.
3. Task analysis stage (concept of action-objective sequence).
4. Modeling stage (requires choosing a model and an architecture).
5. Specification stage (specifications)
   - Needs study for the HMI
   - Conceptual specification
   - Functional specification
   - Syntactic specification
   - Lexical specification

Chapter 3: Models & architectures

1. The Dialog Controller (definition & role)
2. Presentation of the Seeheim model
3. Presentation of the CAP model
4. Presentation of the MVC model
5. Presentation of agent models.

Chapter 4: Ergonomic rules in HMIs

1. Nielsen heuristics
2. Ergonomic criteria of Bastien and Scapin
3. Coutaz's golden rules

Chapter 5: Multi-user interfaces design

1. Comparative study between single-user and multi-user HMI.
2. The UCD method (User-Centric Design)
3. Examples of multi-user interfaces.

Chapter 6: Adaptive Interfaces

1. The Vaudry model.
2. Study of an example: Model with agents.

Chapter 7: Multimodal interfaces and future interfaces

1. Advanced interaction techniques (Augmented Reality, Tangible Interface, 3D projection, Analysis movement)
2. Elements of Visual Programming.

Evaluation method: Exam (60%) , CC (40%)

References:

- Ménadier Jean-Paul, L'interface utilisateur : Pour une informatique conviviale, DUNOD, Informatique et Stratégie, 1991
  Coutaz Joelle, Interface homme-ordinateur : conception et réalisation Dunod-Informatique 1990
- Kolski, C, Ezzedine, H et Abed, M, « *Développement du logiciel : des cycles classiques aux cycles enrichis sous l'angle des IHM* », ouvrage collectif, Analyse et conception de l'IHM, Interaction homme-machine pour les systèmes d'information Vol 1, Hermès, 2001, 250 p, ISBN 2-7462-0239-5, p. 23-49.
  - Drouin, A, Valentin, A et Vanderdonckt, J, « *Les aports de l'ergonomie à l'analyse et à la conception des systèmes d'information* », in Christophe KOLSKI, (ed.), Analyse et conception de l'IHM, Interaction homme-machine pour les systèmes d'information Vol 1, Hermès, 2001, 250 p, ISBN 2-7462-0239-5, p. 51-83.
- David Benyon, Designing Interactive Systems: A Comprehensive Guide to HCI, UX and Interaction Design,Pearson; 3 edition, 2013
- Yvonne Rogers, Helen Sharp & Jenny Preece, Interaction Design: beyond human computer interaction (3rd edition), Wiley, 2011
- Norman DA, The Design of Everyday Things, Basic Books, 2002. Serengul Smith-Atakan The FastTrack to Human-Computer Interaction, (Paperback) Thomson Learning, 2006.

Semester 5: Computer systems
Methodological Teaching Unit (MTU)
Subject: Linear programming
Credits: 4
Coefficient: 2

Teaching goals: The objectives of this module are to make the student aware of the practical importance of linear optimization problems, understanding the underlying theory, and being able to use these techniques in practical problems.

Recommended Prerequisite knowledge: Mathematics, Computer science fundamentals.

Subject Content:

Chapter 1: General introduction

1. History of linear programming
2. Examples of modeling practical problems in the form of a linear program.

Chapter 2: Geometry of Linear Programming

1. Vector spaces, rank of matrix, systems of linear equations
2. Convex set, hyperplane, polyhedron, simplex, extreme point

Chapter 3: Main method of solving a linear program

1. Position of the problem
2. Characterization of the extreme points
3. Optimality at an extreme point
4. Criterion of optimality: formula for increasing the objective function, Opitimality criterion
5. Sufficient condition of existence of unbounded solution
6. Simplex algorithm: improvement of the objective function by passing from one extreme bridge to another, simplex algorithm in matrix form, finiteness of the simplex algorithm simplex algorithm and table
7. Initiation of the simplex algorithm: case of the linear program in normal form, M-method, two-phase method

Chapter 4: Dual methods in linear programming

1. Definitions
2. Dual function increase formula and optimality criterion
3. Sufficient condition of feasible solutions in the primal problem
4. Dual Simplex Algorithm Initiation of the dual simplex algorithm

Evaluation mode : Exam (60%), CC (40%)

References

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

Semester 5: Computer systems
Methodological Teaching Unit (MTU)
Subject: Probabilities and statistics
Credits: 4
Coefficient: 2

Teaching goals: This course is an introduction to the study of simple random models. The objective is to provide the essential tools in the field of probabilities, and also to approach statistical aspects. At the end of this module, the student should be able to calculate the different measures of dispersion in statistics and to perform probabilities based on the laws of probability and to test data using theories of probability.

Recommended Prerequisite knowledge:

Subject Content:
1. Probability spaces
2. Discrete random variables
3. Continuous random variables
4. Characteristic functions
5. Limit theorems
6. Gaussian vectors
7. Simulation
8. Estimators
9. Testing
10. Interval and confidence regions
11. Problems (probabilities)
12. Problems (probabilities and statistics)

Evaluation mode : Exam (60%), CC (40%)
References

- Lecoutre B., Tassi Ph. (1987) Statistique non paramétrique et robustesse Paris : Economica.
- Tassi Ph. (1989) Méthodes statistiques Paris: Economica –
- Tassi Ph., Legait S. (1990) Théorie des probabilités en vue des applications statistiques Paris : Ed. Technip
- Saporta, G., Probabilités, Analyse des données et Statistique, Technip, 2ème édition, 2006
- Jean-Pierre Lecoutre, Statistique et probabilités, Editions Dunod, 2012.
- Yadolah Dodge, Valentin Rousson, Analyse de régression appliquée, Editions Dunod, 2004.

Semester 5: Computer systems
Fundamental Teaching Unit (FTU)
Subject: Compilation
Credits: 5
Coefficient: 3

Teaching goals: Understand how programs are compiled and executed.

Recommended Prerequisite knowledge: Theory of language.

Subject Content:
Chapter 1: introdution (goals) …
Chapter 2 : Compilation

i. Compilator definition
ii. Compilator design

Chapter 3 : lexical analysis
Chapter 4 : Syntax analysis

i. The left most derivation and derivation tree
ii. Ambiguous grammar
iii. programmation language and grammar
iv. Syntax analyzers and there types
v. tools in practice

Chapter 5 : Tpo down parsing

i. The LL(1) parsing (principe)
ii. Parsing table
iii. The LL(1) grammer

Chapter 6 : Bottom up parsing

i. LR parsing (principe)
ii. LR(0) parsing
iii. SLR(1) parsing
iv. LR(1) parsing
v. LALR(1) parsing

Chapter 7 : Syntax- Directed translation

Chapter 8 : Type control

Chapitre 9: Runtime Environnements

Chapter 10 : code generation

Evaluation method: Exam (60%) , CC (40%)

References:

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

Semester 5 : Computer systems
Fundamental Teaching Unit (FTU)
Subject: Software engineering
Credits: 5
Coefficient: 3

Teaching goals: learn to apply an analysis methodology for the software development. In particular, learn object modeling with the universal language UML.

Recommended Prerequisite knowledge: Algorithm, information system, Object Oriented Programming.

Subject Content:
Chapter 1. Introduction

1. Definitions and objectives
2. Principles of Software Engineering
3. Expected Quality of a software
4. Life cycle of a software
5. Model of software life cycle

Chapter 2: Modeling with UML

1. Introduction

Modeling, Model, Object Oriented Modeling, UML in application.

2. General elements and mechanisms
3. The UML diagrams
4. Packages

Chapter 3. UML use case diagram : functional view
Interest and definition, Notation

Chapter 4. UML class and object diagram : static view

1. Class diagram
2. Object diagram

Chapter 5. UML diagrams : dynamic view

1. Interaction diagram (sequence and collaboration).
2. Activity diagram
3. Transition state diagram

Chapter 6. Other concepts and UML diagrams

1. Components, deployment, composite structures.
2. Extension mechanisms: OCL language + profiles.

Chapter 7: Introduction to development methods: (RUP, XP)
Chapter 8: Design patterns and their place in the development process

Evaluation method: Exam (60%) , CC (40%)

References:

1. Bern Bruegge and Allen H. Dutoit, Object-Oriented Software Engineering –using UML, Patterns and Java. Third Edition, Pearson, 2010.
2. G. Booch, J. Raumbaugh, I. Jacbson, “The Unified Modeling Language (UML) Reference Guide”, Addison-Wesley, 1999.

3. G. Booch, J. Raumbaugh, I. Jacbson, “The Unified Modeling Language (UML)User Guide”, Addison-Wesley, 1999.
4. G. Booch et al., “Object Oriented Analysis and design, with application”Addison- Wesley, 2007.
5. Laurent Audibert. Cours UML 2.0, dispobible sur http://www.developpez.com.
6. M. Blaha et J. Rumbaugh. Modélisation et conception orientées objet avec UML 2. 2ème édition. Pearson Education, 2005.
7. Pierre-Alain Muller. Modélisation objet avec UML. Éditions Eyrolles, 2003.
8. Shari Lawrence Pfleeger and Joanne M. Atlee.Software Engineering. Fourth Edition, Pearson, 2010.

Semester 5 : Computer systems
Transversal Teaching Unit (TTU)
Subject: Digital economy and strategic watch
Credits: 2
Coefficient: 1

Teaching goals: the age of information has imposed itself in the 21st century. The aim of this object is to attribute and equips the student with knowledge of two concepts to be used in the near and distant future : Digital economy and strategic watch.

Subject Content:
Chapter 1. Digital economy

1. History and definition
2. Electronc commerce
3. Electronic contract and electronic signature
4. Electronic prospecting and advertising
5. Intellectual and commercial property

Chapter 2. Strategic watch

1. Watch Concepts and Watch Types
2. Watch process models
3. The detailed steps of the watch process
4. Overview of several free monitoring tools for collection, management and dissemination

Chapter 3. Watch and Social media

1. Plan, collect and organize information
2. Reputation monitoring
3. Content curation
4. Particularities of monitoring on social networks
5. Create a (good) LinkedIn profile and build your network
6. Practical workshop: Twitter, LinkedIn, Mention, Tweetdeck

Chapter 3. Mangement of a watch project

1. Manage a watch project
2. Work effectively in team
3. Identifying information sources

Evaluation mode: Exam (100%)

References:

- Système d'information pour le management stratégique : l'entreprise intelligente. Ed. McGraw Hill , Paris, 146 p Ce livre a reçu le Prix Harvard
- Francine Séguin, Taïeb Hafsi et Christiane Demers, Le management stratégique, de l'analyse à l'action. Les Éditions Transcontinental, 2008.
- Veille stratégique : Comment ne pas être noyé sous les informations. Économies et Sociétés, Série Sciences de Gestion, n°2/1998, p.159-177. LESCA, H. (2001)
- Veille stratégique : passage de la notion de signal faible à la notion de signe d'alerte précoce. Colloque VSST 2001, Barcelone oct., Actes du colloque, tome 1. LESCA, H. CARON, M-L (1995) - Veille stratégique : créer une intelligence collective au sein de l'entreprise. Revue Française de Gestion, sept. - oct. , p.58-68. LESCA, H RAYMOND, L. (1993).
- Expérimentation d'un système expert pour l'évaluation de la Veille Stratégique dans les PME. Revue Internationale PME, vol.6 n°1 p.49-65. (Québec, Canada) Site internet http://www.veille-strategique.org

Semester 6 : Computer systems
Fundamental Teaching Unit (FTU)
Subject : Mobile applications
Credit : 5
Coeffecient : 3
Teaching goals : The aim of this subject is to equip the student with basic knowledge about computer application development in mobile environment. With the arrival of smartphones, mobile applications are ubiquitous either in the Client side (BtoC), Provider (BtoB) or Collaborator (BtoB). The goal is also to learn to deevlop in the Android environment and its specificities of embedded development on smartphones.
Prerequisites : Good notions on Java, Javascript and XML
Subject content :
Chapter 1 : Mobile applications

1. Introduction
2. Mobile operating systems
3. Types of mobile applications

Chapter 2 : Android platform

1. Android platform presentaion
2. Fundamental components of an Android application
3. Types of mobile applications
4. The Android SDK
5. Installation and configuration of tools
6. Create an Android emulator
7. The first Android app

Chapter 3 : Activities and resources

3. Introduction
4. Notion of activity
5. Lifecycle of an activity
6. Resources
7. Organization of resources
8. Use of resources
   - Character strings
   - Drawables
   - Styles
   - Animations

Chapter 04: GUIs and Widgets

1. Creation of graphical interfaces
2. Manage events on widgets

Chapter 05: Menus and dialog boxes

1. User management of the app
   a) Options menu
   b) Context menus
2. Dialogs

Chapter 06: AndroidManifest.xml and communication between components

1. The AndroidManifest.xml file
2. Communication between components
   a) Explicit intents
   b) Implicit intents

c) Resolution of implied intents

Chapter 07: Databases with SQLite

Chapter 08: A Simple Application

Evaluation mode : Exam (60%), CC (40%)

References

Créez des applications pour Android - OpenClassrooms
https://openclassrooms.com/courses/creez-des-applications-pour-android

Développement Android - Jean-Francois Lalande
http://www.univ-orleans.fr/lifo/Members/Jean-Francois.Lalande/enseignement/android/cours-android.pdf

Semester 6 : Computer systems
Fundamental Teaching Unit (FTU)
Subject : IT Security
Credit : 5
Coeffecient : 3
Teaching goals : This subject allows students to acquire skills to ensure the security and proper functioning of computer systems.
Prerequisites : Algorithmics, Programming techniques
Subject content :

Chapter 1 : Introduction to IT Security

1.1 Definitions: Security, Dependability, ...
1.2 Main IT Security concepts : vulnerability, - threat, - countermeasure, risk, ...
1.3 IT security objectives : Confidentiality, Integrity, Availability, Non-repudiation, Authentication, ...
1.4 IT threats :

- What is an attack?
- Definitions: Virus - Worm - Trojan Horse - Spyware
- Origin of attacks
- Who can be targeted?
- Stages of an attack
- The different taxonomies of attacks
- The different types of attacks: - Network attacks - System attacks - Password attacks - Website attack - Application attack. - Ways to launch an attack

1.5 Defense methods : - Anti-virus, - Firewalls, - Private networks, - Intrusion detection, etc...

Chapter 2 : Introduction to Cryptography

2.1 Vocabulary and definitions : - Cryptology, - Cryptography, Cryptogram, - Cryptanalysis, etc...
2.2 History of cryptography
2.3 Classical Cryptography : - Substitution algorithm: Caesar cipher, VIGENERE cipher.
- Transposition algorithm: the Assyrian technique.
2.4 Modern Cryptography : - Symmetric cryptography: Principles, DES and AES algorithms
- Asymmetric Cryptography: Principles, RSA Algorithm
2.5 Hash Functions : - Principles - MD5 and SHA-1 algorithms.
2.6 The digital signature
2.7 Digital certificates
2.8 Certification authorities and PKI

Evaluation mode : Exam (60%), CC (40%)

References

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

Semester 6 : Computer systems
Fundamental Teaching Unit (FTU)
Subject : Artificial Intelligence
Credit : 5
Coeffecient : 3
Teaching goals : The main objective of this subject is to allow a student to know the fundamental concepts of AI. It consists of allowing more interests to the contribution of AI in problems resolution, which is actually based on reasoning approaches rather than classical computational techniques.
Prerequisites : Mathematical logic, Algorithmics
Subject content :
Chapter 1 : Beginning of AI

1.1 History : Beginning of AI, AI and types of problems, Difference between AI and classical computational techniques
1.2 Turing test
1.3 AI applications

Chapter 2 : Expert systems

2.1 Definition and role
2.2 Expert system architecture

Chapter 3 : Expert system functioning

3.1 Knowledge and Knowledge representation formalisms
3.2 Production rules-based systems
3.3 Inference engine functioning

Chapter 4 : Expert system developping appraoch

4.1 Expert system developping process
4.2 Examples of expert systems : Dendral, Mycin, Prospector

Evaluation mode : Exam (60%), CC (40%)

References :

- Manuel d’intelligence artificielle, Louis Frécon and Okba Kazar, Edition PPUR, ISBN : 978-2-88074-819-7, 2009
- Ganascia, Jean-Gabriel. L’intelligence artificielle, Flammarion, 1993
- Bratko, Programmation en Prolog pour l’intelligence artificielle, 2001
- J.M. Alliot et T.Schiex, Intelligence Artificielle et Informatique Theorique, Cepadues Editions, 1993.
- N. Nilsson, Artificial Intelligence: A New Synthesis, Morgan Kaufmann, 1998.
- S. Russell, P. Norvig, Artificial Intelligence: A Modern Approach, 2nd edition, 2002

Semester 6 : Computer systems
Fondamental Teaching Unit (FTU)
Subject: Semi-structured Data
Crédits : 5
Coefficient : 3
Objectives: the goals of this subject is to allow the student to become familiar with unstructured data to be used in web-oriented applications.
Prerequisites : Programming languages
Contents :

1. Context and problematic

- 1.1. Databases
- 1.2. Multimédia and document
- 1.3. Hypermédia, Internet and Web
- 1.4. Problematic

2. Documents and hyper multimédias-documents

- 2.1. Documents
- 2.1.1. Introduction
- 2.1.2. Spécific documents Modelling
- 2.1.3. Classes documents Modelling
- 2.2. Hyper documents
- 2.3. The multimédias contents

3. XML core

- 3.1. Introduction to XML
- 3.2. Basic XML Structure
- 3.3. Nominal domains
- 3.4. XML Schémas

4. XML Galaxy

- 4.1. XML path : XPATH
- 4.1.1. Principles
- 4.1.2. Axes
- 4.1.3. Filters
- 4.1.4. prédicates
- 4.2. Style sheets and XSL processing
- 4.3. XML Applications: RDF, SVG, ...
- 4.4. Processing XML: DOM and SAX
- 4.5. Pointers: XPOINTER
- 4.6. Links: XLINK

5. XML and semi-structured databases

- 5.1. Semi-structured data and XML
- 5.2. Query languages
- 5.3. XML Databases

6. XQUERY and databases

- 6. 1. Semi-structured data and XML
- 6.1.1. Semi-structured databases
- 6.1.2. Query languages
- 6.1.3. XML Databases
- 6. 2. XQUERY
- 6.2.1. XQuery Syntax
- 6. 2.1.1. XQuery expressions
- 6.2.1.2. Functions and operations
- 6.2.2. Functions usage examples

Evaluation mode : Exam (60%), CC (40%)

References

- CHAUDHRI Akmal, RASHID Awais, ZICARI Roberto " XML Data Management- Native XML and XML Enabled Database Systems " , ADDISON WESLEY , 2003 , ISBN 020184452 4, 641 pages
- Michard A., "XML : langage et application", EYROLLES, 2001, 499 pages , ISBN: 2-212-09206-7
- GARDARIN Georges."Bases de données Objet et Relationnel", EYROLLES, 2001

Semester 6 : Computer systems
Methodological Teaching Unit (MTU)
Subject : Project
Credit : 6
Coeffecient : 3
Teaching goals : The main objective of this subject is to entrust for a group of students to analyse, design and implement a computer application.
Prerequisites : Thre years Computer systems Bachelor curriculum
Subject content :
The Bachelor degree project covers one or more topics covered in the CS curriculum. It is supervised by a teacher from the department. It can be carried out in a company (internship) or at the department.

Evaluation :
The evaluation is done by a jury composed of three (03) teachers, the president, the examiner and the supervisor.

The evaluation includes :

- The thesis (or internship report) : 07 pts
- Application : 07 pts
- Answers to questions : 06 pts

The final score equals to the average of scores given by each jury member.

There is no defense. The consultation is carried on the manuscript and the application (Design and realization).

The thesis (or the internship) includes the essential of the work and it must not exceed 30 pages.

Assessement scale : Report (35%), Application (35%), Answers (30%)

Semester 6 : Computer systems
Methodological Teaching Unit (MTU)
Subject : Scientifc writing
Credit : 2
Coeffecient : 1
Teaching goals : Learn the scientific writing technique in order to allow the students to comment and publish their research work results.
Prerequisites : Knowledges about scientific works.
Subject content :

- **Introduction**
- **Before writing**
  a. Ideas organization and writing plan
  b. Prepare the writing method
  c. Select the layout tools
  d. Versions control systems
- **Bibliographic references**
  a. Get the references
  b. Build the bibliography
  c. Avoid the plagiarism
- **Writing**
  a. Structure
  b. Content
  c. Style
  d. Presentation

Evaluation mode : CC (100%)

References

- BRUYERE, V. Comment bien rédiger. Exposé ; à l'intention des étudiats de la première année licence en informatique, Université de Mons-Hainaut, 2006.
- VALDURIEZ, P. Some Hints to Improve Writing of Technical Papers. Ingénierie des Systèmes d'informations 2,3 (1994), 371–375

Semester 6 : Computer systems
Methodological Teaching Unit (MTU)
Subject : Create and develop a Startup
Credit : 2
Coeffecient : 1

Teaching goals : This course has as objective to help the student to structure, strart or develop his entrepreneurial project. The aim is also to develop the entrepreneurship learning for the student. It also consists of developping the entrepreneurial creativity by giving a value of the project idea using "business model" project type. The subject is essentially divided into four main points:

- Motivate the student to create a Startup and give him all the resources he need to begin his business project
- Master the tools of formalizing and implementing the Startup project
- Turning good ideas into business model creation
- Adapt the student to the startup ecosystem and culture

Prerequisites : Programming language, Entreprise concept.

Subject content :

- **Chapter 1 : Entrepreneurial profile and motivations**
  a. The creation of a Startup
  b. The entrepreneur posture
  c. How to find an idea
  d. The Startup ecosystem
  e. Start alone or with others
  f. Growth and sales
  g. Errors, failures and best practices
- **Chapter 2 : From idea to market**
  a. Customer and product development
  b. Building a formalizing economic model
  c. The PITCH art
- **Chapter 3 : From market to growth**
  a. The growth hacking
  b. Fundraising and valuation
  c. The crowdfunding: marketing and financial leverage
- **Chapter 4 : Adminstration and management**
  a. The starting pack : legal/ social/ tax/ VAT
  b. The tools to manage, develop and communicate on its activity
  c. Market acces and Sales, Business development, Web marketing, Perfromance mangement
  d. Strategic intelligence

Evaluation mode : CC (100%)

References

- Robert Papin, La création d'entreprise, Création, reprise, développement, 16e édition - *Collection : Hors collection, Dunod,* 2015.
- Eric Ries, Lean Startup : Adoptez l'innovation continue, Éditeur : PEARSON, 2015.
- Vincent Ydé, Créer son entreprise : du projet à la réalité, Éditeur : VUIBERT, 2009.