PÉRIODE D'ACCRÉDITATION : 2016 / 2021

UNIVERSITÉ PAUL SABATIER

SYLLABUS MASTER

Mention Informatique

M1 Computer Science for Aerospace

http://www.fsi.univ-tlse3.fr/
http://m1.deptinfo.fr/

2016 / 2017

5 JUILLET 2017

SUMMARY OF THE CONTENT

DIAGRAM OF LINKS BETWEEN BACHELOR AND MASTER DEGREES . . . 3

PRESENTATION . . . 4

- PRESENTATION OF DISCIPLINE AND SPECIALTY . . . 4
  - Discipline Informatique . . . 4
  - Speciality . . . 4
- PRESENTATION OF THE YEAR OF M1 Computer Science for Aerospace . . . 4

CONTACTS SECTION . . . 5

- CONTACT INFORMATION CONCERNING THE SPECIALTY . . . 5
- CONTACT INFORMATION CONCERNING THE DISCIPLINE . . . 5
- CONTACT INFORMATION FOR THE DEPARTMENT : FSI.Info . . . 5

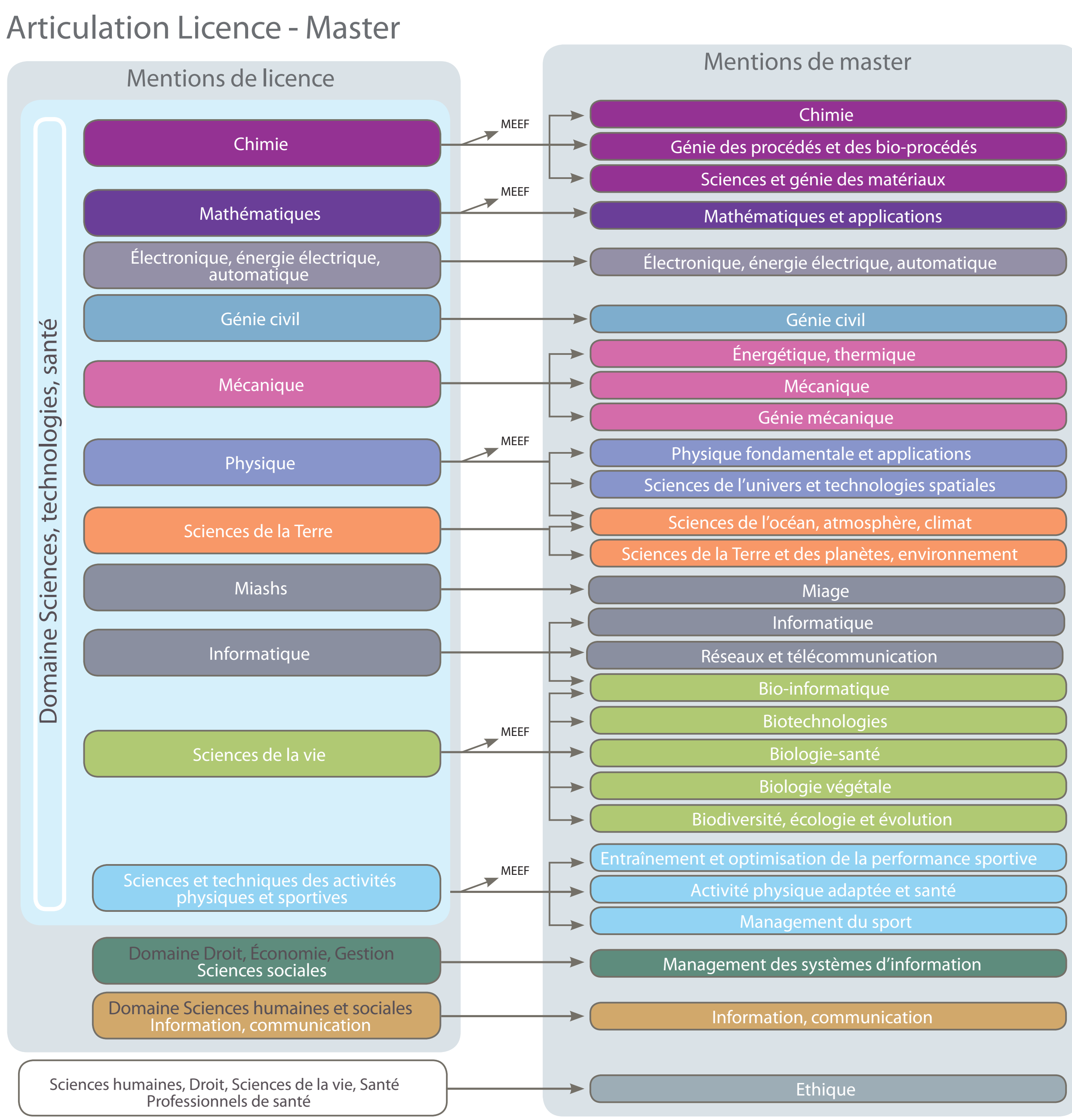
Table summarizing the modules that make up the training program . . . 6

LIST OF THE MODULES . . . 7

GLOSSARY . . . 26

- GENERAL TERMS . . . 26
- TERMS ASSOCIATED WITH DEGREES . . . 26
- TERMS ASSOCIATED WITH TEACHING . . . 26

DIAGRAM OF LINKS BETWEEN BACHELOR AND MASTER DEGREES



MEEF : cf. page 10. Projet métiers de l'enseignement

PRESENTATION

PRESENTATION OF DISCIPLINE AND SPECIALTY

DISCIPLINE INFORMATIQUE

Computer science is nowaday at the core of many societal, industrial and scientific domains. The aim of the computer science master program at the university Paul Sabatier is to give student a in-depth expertise in several domain of computer science.
In the first year of this master, a set of common skills is delivered as the basis for a progressive specialization.
In the second year of this master, strong specialization year, theoretical and technological high-level training is offered to students, allowing them to access the many opportunities in the computer science industry but also to continue their doctoral studies.
The computer science master program is declined around the following thematic areas :
- Information processing and infrastructure
- Software engineering as a set of concepts, methods and development tools.
- Manipulation of content from different points of view : analysis / synthesis of information, structuring and retrievial of information, integrating the problem of massive data.
- Representation and processing of knowledge in artificial intelligence, with links toward robotic.
- Man machine interaction, with ergonomic and cognitive constraints relating thereto.

SPECIALITY

The aim of the Computer Science for Aerospace (CSA) Master Degree is to educate future computer scientists and managers, within an international working environment, selecting talented students from all around the world. It aims to provide specialist high-level core knowledge and a unique specialty in computer science for aerospace, encompassing both hardware and software skills.

PRESENTATION OF THE YEAR OF M1 COMPUTER SCIENCE FOR AEROSPACE

Students enroll in a two-year curriculum covering the fundamental notions of Critical systems, Embedded systems, Safety, Security, Certification, Interactive systems, Dedicated architectures, Networks and systems, Real-time systems, Image analysis, Artificial intelligence, Information systems and Databases.

CONTACTS SECTION

CONTACT INFORMATION CONCERNING THE SPECIALTY

PERSON IN CHARGE OF TEACHING AFFAIRS OF M1 COMPUTER SCIENCE FOR AEROSPACE

KOUAME Denis
Email : denis.kouame@irit.fr

OBER Ileana
Email : Ileana.Ober@irit.fr Téléphone : 05 61 55 74 23

PAULIN Mathias
Email : Mathias.Paulin@irit.fr Téléphone : 05 61 55 83 29

PIERSON Jean-Marc
Email : Jean-Marc.Pierson@irit.fr Téléphone : 05 61 55 72 26

SECRETARY OF STUDENT AFFAIRS OF

PIERUCCIONI Corinne
Email : corinne.pieruccioni@univ-tlse3.fr Téléphone : 05.61.55.86.39

Université Toulouse III - Paul Sabalier
F.S.I. - Division Formation
U3 - Porte 112 - 1er étage
118 route de Narbonne
31062 TOULOUSE cedex 9

CONTACT INFORMATION CONCERNING THE DISCIPLINE

PERSON IN CHARGE OF THE DISCIPLINE INFORMATIQUE

KOUAME Denis
Email : denis.kouame@irit.fr

PAULIN Mathias
Email : Mathias.Paulin@irit.fr Téléphone : 05 61 55 83 29

CONTACT INFORMATION FOR THE DEPARTMENT : FSI.INFO

HEAD OF DEPARTMENT

CROUZIL Alain
Email : alain.crouzil@irit.fr Téléphone : 05 61 55 69 28

DEPARTMENT SECRETARY

LESTRADE Colette
Email : lestrade@adm.ups-tlse.fr Téléphone : 05 61 55 81 58

Université Paul Sabalier
1TP1-14
118 route de Narbonne
31062 TOULOUSE cedex 9

TABLE SUMMARIZING THE MODULES THAT MAKE UP THE TRAINING PROGRAM

page	Code	Title of the module	ECTS	Mandatory Optional	Cours	TD	TP	Projet	Stage
First semester									
8	EMINE1BM	ADVANCED ALGORITHMIC	5	O	12	16	12		
9	EMINE1CM	OBJECT MODELLING	5	O	10	20	10		
10	EMINE1DM	PARALLELISM / DISTRIBUTED SYSTEMS	5	O	12	16	12		
11	EMINE1EM	LANGUAGE THEORY	5	O	16	14	10		
12	EMINE1FM	SCIENTIFIC COMPUTING	4	O	14	10	8		
13	EMINE1GM	COMPUTER-HUMAN INTERACTIVE SYSTEMS	3	O	8	8	8		
15	EMINE1VM	ANGLAIS	3	O		24			
14	EMINE1TM	STAGE FACULTATIF	3	F					0,5
Second semester									
16	EMINE2AM	BUSINESS	3	O		24			
17	EMINE2BM	ARCHITECTURE, SYSTEMS, NETWORKS	6	O		26	22		
18	EMINE2CM	SECURITY	3	O		12	12		
19	EMINE2DM	INTRODUCTION TO EMBEDDED SYSTEMS	3	O		14	10		
	EMINE2EM	SOFTWARE TOOLS - PROJECT	3	O					
20	EMINE2E1	Software Tools - Project (classroom)			6				
	EMINE2FM	RESEARCH INITIATION	3	O					
21	EMINE2F1	Research Initiation (classroom)			6				
22	EMINE2GM	ADVANCED OPTIMIZATION	3	O	6	10	8		
23	EMINE2HM	INTRODUCTION TO DISTRIBUTED DATABASES	3	O	10	8	6		
Choose 1 module among the following 2 modules :									
24	EMINE2VM	ANGLAIS	3	O		24			
25	EMINE2YM	FRANÇAIS GRANDS DÉBUTANTS	3	O		24			

LIST OF THE MODULES

UE	ADVANCED ALGORITHMIC	5 ECTS	1^{st} semester
EMINE1BM	Cours : 12h , TD : 16h , TP : 12h		

TEACHER IN CHARGE OF THE MODULE

BANNAY Florence
Email : Florence.Bannay@irit.fr Téléphone : 05 61 55 63 30

LEARNING GOALS

Many present-day situations require an optimal solution. In order to find such a solution the computational tool should use efficient data representation and efficient algorithms. The user may want to obtain either integer-valued or real-valued solutions. Some kinds of problems can be solved optimally with polynomial-time algorithms, for other ones the best solution is too difficult to compute and the user should content themselves with the results returned by incomplete algorithms.

SUMMARY OF THE CONTENT

Introduction to combinatorial optimization problems with examples
1. Algorithmic complexity and Efficient data structures
a) temporal and spatial complexity
b) data-structure for priority handling (B heap, binomial heap)
c) data-stracture for information seeking (B-tree and balanced tree)
2. Optimisation problems with polynomial-time algorithms
a) Flow theory (transport network, min cut theorem, Ford-Fulkerson algorithm, min cost flows)
b) Linear Programming (graphical solution, matrice resolution, simplex, dual)
3. Meta-heuristics
a) neighbourhood search
b) genetic algorithms
Conclusion : branch and bound algorithm
Practical work and projects : 1) implementation of a kd-tree applied to image generation, 2) modeling small Flow problems and LP problems, 3) implementation of a local search algorithm applied to traveling salesman problem

PREREQUISITES

Graph theory, Data structures

REFERENCES

Introduction to Algorithms, third edition. By Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein. MIT Press.
Talbi, E. Metaheuristics - From Design to Implementation Wiley, 2009.

KEYWORDS

amortized complexity, heap, B-tree, kd-tree, Simplex, Flow, Meta-heuristics, Local Search, Genetic algorithms

UE	OBJECT MODELLING	5 ECTS	1^{st} semester
EMINE1CM	Cours : 10h , TD : 20h , TP : 10h		

TEACHER IN CHARGE OF THE MODULE

OBER Ileana
Email : Ileana.Ober@irit.fr Téléphone : 05 61 55 74 23

LEARNING GOALS

This lecture addresses the modeling and design of complex software.
The lecture starts with an overview of the issues specific to the development of complex software.The topics addressed in this lecture are : software modeling, its use in the context of a development process, the specification of constraints related to the modeled systems in order to ensure their coherence and the design using design patterns.
This lecture focuses on the modeling language UML and on its use on concrete settings through advanced modeling exercises and practical sessions, that help the students go beyond the syntactic aspects of the language and help them perceive the need for including the modeling steps in the framework of an overall development process.

SUMMARY OF THE CONTENT

Managing complex software, basic techniques for managing complexity (decomposition vs. abstraction)
Development process (using a clearly identified development process along the modeling and design lectures)
UML modeling
– requirements modeling
– structural modeling
– introduction to constraints specification using OCL
– behaviour modeling
Introduction to model transformation
Pattern based design
– introduction to design patterns
– overview and classification of design patterns
– main structural and behavioral patterns : Strategy, Adapter, Facade, Observer, Decorator...
– introduction to creational patterns

PREREQUISITES

Object oriented programming, basic knowledge of UML (class and sequence diagrams)

REFERENCES

B. Bruegge. OO Software Engineering Using UML, Patterns and Java, Pearson, 2009
J Warmer, A Kleppe The OCL, Addison Wesley 2003
E. & E. Freeman, Head First Design Patterns, O'Reilly, 2005

KEYWORDS

modeling, design, development method, constraint specification, OCL, model transformation, design pattern, software flexibility

UE	PARALLELISM / DISTRIBUTED SYSTEMS	5 ECTS	1^{st} semester
EMINE1DM	Cours : 12h , TD : 16h , TP : 12h		

TEACHER IN CHARGE OF THE MODULE

BAHSOUN Jean Paul
Email : bahsoun@irit.fr Téléphone : 0561558211

LEARNING GOALS

Construction of reliable and performant models
Properties classification as safety and liveness
Study of several models for parallelism and distribution
cache data coherency in a shared-memory architecture
performance analysis of shared-memory parallel programs

SUMMARY OF THE CONTENT

Foundation and principles of parallel and distributed models
Data parallelism
cache data coherency in a shared-memory architecture
performance analysis of shared-memory parallel programs : estimation of the synchronisation/communication costs and speedup of various algorithms
Pallel and Distributed modelling
Synchronous and asynchronous models
Modelling parallelism with Petri nets
Compostion with process algebra
Shared Variables, synchronization conditions and message passing
Distributed Algorithms for broadcast, election, tremination, consensus, token, fragmentation and replication

PREREQUISITES

Consurrent programming, processes, threads, synchronisation, shared variables

REFERENCES

Fundamentals of Parallel Multicore Architecture, Y. Solihin (Chapman and Hall/CRC)
Principles of Concurrent and Distributed Programming, Addison-Wesley, 2006.
Communication and Concurrency, Prentice Hall , 1995, R. Milner

KEYWORDS

Parallel architectures, Parallel models, Distributed models, Performance, Data coherece, Synchronisation

UE	LANGUAGE THEORY	5 ECTS	1^{st} semester
EMINE1EM	Cours : 16h , TD : 14h , TP : 10h		

TEACHER IN CHARGE OF THE MODULE

BONENFANT Armelle
Email : bonenfant@irit.fr Téléphone : 05 61 55 6360

LEARNING GOALS

Acquire basic methods of definition and treatment of program languages : grammar, semantics, interpretation, compilation and optimization. Understand the structure of a compiler. Implement a translator to transform an external representation into a generated code for a program language considering its specification, modelisation, and data manipulation.

SUMMARY OF THE CONTENT

1. General principle on compilation, interpretation vs compilation
2. Abstract and concrete syntax, symbol table
3. Descendant syntactic analysis : LL grammar, descendant recursive procedures
4. Translation and code generation (intermediate language of quadruplets)
5. Optimization Search Strategies for Relational Languages
– Algebraic Restructuring Methods
– Deterministic and Randomized Search Strategies
6. Ascendant analysis : principles, LR grammars, ascendant code generation

PREREQUISITES

Languages, Automatons, Relational Databases

REFERENCES

Compilers : Principles, Techniques, and Tools 2nd Edition, A.V. Aho, M. S. Lam, R. Sethi& J.D. Ullman, Ed. Addison Wesley
Relational Databases and Knowledge Bases, G. Gardarin & P. Valduriez, Ed. Addition Wesley

KEYWORDS

Syntactic analysis - Translation - Optimization

UE	SCIENTIFIC COMPUTING	4 ECTS	1st semester
EMINE1FM	Cours : 14h , TD : 10h , TP : 8h		

TEACHER IN CHARGE OF THE MODULE

MOUYSSET Sandrine
Email : sandrine.mouysset@irit.fr

LEARNING GOALS

Learn mathematical tools to represent and solve physics systems.
Basis of optimization and algorithm will be introduced.
Moreover, optimization problem will be applied on real case.

SUMMARY OF THE CONTENT

Optimization :

1. Multivariate calculus : partial differentiation, contour lines, gradient, hessian, singular points
2. Intoduction to convex analysis and optimization, numerical algorithms
3. Problems with root mean square errors
4. Solving linear systems : direct and iterative methods

PREREQUISITES

Linear algebra, numerical methods

REFERENCES

Introduction à l'analyse numérique matricielle et à l'optimisation, P. Ciarlet, Dunod
Introduction à l'optimisation et au calcul semi-différentiel, M. Delfour, Dunod

KEYWORDS

Optimization, linear systems, numerical algorithms

UE	COMPUTER-HUMAN INTERACTIVE SYSTEMS	3 ECTS	1st semester
EMINE1GM	Cours : 8h , TD : 8h , TP : 8h		

TEACHER IN CHARGE OF THE MODULE

MARTINIE Celia
Email : Celia.Martinie-De-Almeida@irit.fr Téléphone : 0561557707

PALANQUE Philippe
Email : Philippe.Palanque@irit.fr Téléphone : 0561556965

LEARNING GOALS

Know the basic principles of Human-Computer Interaction & User-Centred Design
Master the principles of the of design and modeling of interactive systems. Make students able to propose a software architecture for the interactive part of computer programs :

- Define a software architecture for interactive applications ensuring usability, modifiability and reliability ;
- Design interactive applications using the Model View Controller design pattern and know how to connect it to the software achitecture ;
- Model all the components of an interactive application (input, output and behavioural aspects) ;
- Demonstrate the reliability of interactive applications (through testing and formal verification of properties) ;
- Deploy those concepts and principles within an event-driven programming environment (NetBeans) ;

SUMMARY OF THE CONTENT

0. Introduction to HCI and the principles of User Centred Design
1. Introduction to the architectural principles of interactive systems (Seeheim and ARCHI software architectures)
2. Presentation of the basic principles underlying the modelling of interactive systems using extended automata (Augmented Transition Networks - Wood 70)
3. Presentation of MVC design pattern, its use for interactive systems development, its expliotation in Java programming and its integration within ARCH software architecture ;
4. Model-based implementation of interactive systems within an event-driven programming environment ;
5. Description of generic and specific properties of interactive systems and their verification on models ; How to take into account within the same framework usability ans reliability ;
7. Implementation of valisation principles : properties verification, definition and implementation of software tests for interactive systems.

PREREQUISITES

Object-Oriented Programming (Java), Java SWING class library, event-driven programming, State-based modeling using Automata

REFERENCES

Buxton, W., 1990. A three-state model of graphical input. IFIP TC13 Conference on HCI, North-Holland Publishing Co., 449-456
L. Bass, P. Clements, R. Kazman, Software Architecture in Practice, (3rd edition), Addison-Wesley, 2012.

KEYWORDS

Foundation of HCI, Interactive systems modeling, reliability, usability, verification and test

UE	STAGE FACULTATIF	3 ECTS	1^{st} semester
EMINE1TM	Stage : 0,5 mois minimum		

UE	ANGLAIS	3 ECTS	1^{st} semester
EMINE1VM	TD : 24h		

TEACHER IN CHARGE OF THE MODULE

CHAPLIER Claire
Email : claire.chaplier@univ-tlse3.fr

UE	**BUSINESS**	**3 ECTS**	**2nd semester**
EMINE2AM	TD : 24h		

TEACHER IN CHARGE OF THE MODULE

MOTTAY Didier
Email : didier.mottay@univ-tlse3.fr Téléphone : 05 61 55 64 14

UE	ARCHITECTURE, SYSTEMS, NETWORKS	6 ECTS	2^{nd} semester
EMINE2BM	TD : 26h , TP : 22h		

TEACHER IN CHARGE OF THE MODULE

LAVINAL Emmanuel
Email : Emmanuel.Lavinal@irit.fr

LEARNING GOALS

Introducing the contraints of embedded systems, and presenting the architectures, systems and networks used for their implementation.

SUMMARY OF THE CONTENT

Computer architectures :
- constraints of embedded systems
- processors, architectures, multicore chips, microcontrollers, specialized architectures (DSP, GPU, ...), systems on chip
- input/output programming, embedded software

Operating systems :
- basic principles
- cooperating operating systems
- lightweight operating systems

Networks :
- architectures and communication protocols fundamentals
- introduction to industrial networks
- sensor networks
  - introduction and applications
  - routing and MAC protocols
  - UAV-aided networks

PREREQUISITES

- Know how a processor, bus and memory work. Assembly programming.
- Operating system principles

REFERENCES

Operating Systems. Tanenbaum. Pearson
Computer Networking : A Top-Down Approach. James F. Kurose, Keith W. Ross. Pearson

KEYWORDS

microcontrollers, systems on chip, embedded software, lightweight operating systems, sensor networks

UE	SECURITY	3 ECTS	2^{nd} semester
EMINE2CM	TD : 12h , TP : 12h		

TEACHER IN CHARGE OF THE MODULE

TRUILLET Philippe
Email : Philippe.Truillet@irit.fr

LEARNING GOALS

The purpose of this lecture is to understand the issues related to cybersecurity industrial systems, to learn the basics about identifying weaknesses in systems, and learn recommendations and methodology related to cybersecurity and finally understand the key points to consider in the industrial system design phase

SUMMARY OF THE CONTENT

Main concepts of industrial systems
Definition and issues of cybersecurity and key concepts
Duality of operation and cyber security
Vulnerabilities and main attack vectors
Main principles of deploying a cybersecurity project
Panorama standards, benchmarks and standards
Introduction to cryptography ; encryption, hashing, signature and role of these mechanisms
Good practices in cybersecurity

PREREQUISITES

Computer networks concepts.

REFERENCES

Cybersecurity & Cyberwar. P. Singer et al Oxford Univ Press, 2014
Protecting Critical Infrastructures from Cyber Attack & Cyber Warfare, T. Johnson, CRC Press, 2015
Introduction to Computer Networks and Cybersecurity, Ch-H Wu, CRC Press, 2013

KEYWORDS

industrial systems, cybersecrity, computer networks

UE	INTRODUCTION TO EMBEDDED SYSTEMS	3 ECTS	2nd semester
EMINE2DM	TD : 14h , TP : 10h		

TEACHER IN CHARGE OF THE MODULE

OBER Ileana
Email : Ileana.Ober@irit.fr Téléphone : 05 61 55 74 23

LEARNING GOALS

Understand the constraints specific to embedded software
Use asynchronous modeling and programming
Make effective use of modeling and development environments for embedded and real-time systems

SUMMARY OF THE CONTENT

1. Overview of the main characteristics of embedded systems using a case study. Functional and non-functional requirements ; hardware / software architecture (codesign), software development, execution models (synchronous / asynchronous), programming concepts and languages
2. Requirements and refinement via SysML. Introduction to asynchronous specification formalisms
3. Asynchronous modelling Timed automata, Petri-nets
4. Asynchronous programming

PREREQUISITES

Basic knowledge in modeling and in concurrent systems

REFERENCES

Marwedel, Peter Embedded System Design. Embedded Systems Foundations of Cyber-Physical Systems. Springer 2011
Hassan Gomaa Software Modeling and Design. UML, Use Cases, Patterns, and Software Architectures. Cambridge University Press 2011

KEYWORDS

execution models, SysML, asynchronous specification, embedded and real-time systems

UE	SOFTWARE TOOLS - PROJECT	3 ECTS	2^{nd} semester
Sous UE	Software Tools - Project (classroom)		
EMINE2E1	Cours : 6h		

LEARNING GOALS

Knowing how to participate in the development of an IT project, with methodology.
Knowing teamwork.
Getting the practical and methodology tools that will be used in an optional internship.

SUMMARY OF THE CONTENT

The course presents the basic principles of implementation of IT projects.
The bulk of the lecture is based on the realization by a group of students of a significant project proposed by a client. Customers are faculty members of the IRIT laboratory, students in Computer Science or partners from industry.
The project consists of a study or software development. The groups choose the projects, preferably related to their course. The assignment of a project to a group is left to the client's initiative. Labs are given to help the implementation of the concepts discussed during the course section. Student groups perform a delivery of their project with their client, and they present their work in an oral presentation.
Course Outline :
IT projects (nature and issues, actors and roles, essential elements of Project Management)
Quality (content of a quality plan, quality requirements, norms and standards)
Project organization (definition of a development approach, possible approaches to a new development, maintenance)
Visibility (project tracking, summary)

REFERENCES

ISO/IEC DTR 29110-5-6-2 NF X50-120

KEYWORDS

Software project, life cycle, quality, organization, monitoring, assessment

UE	RESEARCH INITIATION	3 ECTS	2nd semester
Sous UE	Research Initiation (classroom)		
EMINE2F1	Cours : 6h		

LEARNING GOALS

Discover the scientific research
Discover the work of a researcher in Computer Science
Deepen knowledge about a research topic
Learn teamwork

SUMMARY OF THE CONTENT

The courses cover the following topics :
Introduction to research (presentation of scientific societies, organization and evaluation of research)
principles of standardization and certification
synthetic and very fast presentation of a research subject with the technique called "Elevator Pitch"
principles of writing research papers (state of the art, integration of citations and references)
A tutorial session is devoted to bibliographic research.
Research work in groups of 3 or 4 students supervised by a researcher, allows to implement the knowledge acquired during the course and practical sessions. This work consists of analyzing a subject of research, conducting a literature search, to summarize the state of the art and to produce the report in the form of a research paper.

REFERENCES

http ://www.acm.org/

KEYWORDS

scientific research, state of the art, writing papers

UE	ADVANCED OPTIMIZATION	3 ECTS	2nd semester
EMINE2GM	Cours : 6h , TD : 10h , TP : 8h		

TEACHER IN CHARGE OF THE MODULE

MENGIN Jérôme
Email : mengin@irit.fr

LEARNING GOALS

To introduce models and algorithms that can be used to solve hard combinatorial optimization problems that can be encountered in a variety of domains, ranging from the management and efficient use of scarce resources to increase productivity or the design of communication network, to e.g. graph theory and Artificial Intelligence.

SUMMARY OF THE CONTENT

- Models and algorithms for solving hard decision and combinatorial optimization problems
  - integer linear programing
  - boolean variables and SAT
  - finite domain constraints
- The NP / NPO problem classes, NP-completeness ; application to cryptography
- Branch & bound methods - complete and incomplete ones
- Polynomial approximations of solutions of NP-complete problems
- Lab sessions :
  - Modeling and solving a large-scale problem using MILP and CSP tools
  - Coding a branch & bound search method

PREREQUISITES

Fundamentals of algorithmics and graph theory

REFERENCES

* S Dasgupta, C Papadimitriou, Umesh V. Vazirani : Algorithms. McGraw-Hill 2008
* C Papadimitriou, K Steiglitz : Combinatorial Optimization : Prentice-Hall 1982
* M. R. Garey, D S. Johnson : Computers and Intractability. W. H. Freeman 1979

KEYWORDS

Combinatorial optimization, complexity theory, integer programming, constraint programming

UE	INTRODUCTION TO DISTRIBUTED DATABASES	3 ECTS	2nd semester
EMINE2HM	Cours : 10h , TD : 8h , TP : 6h		

TEACHER IN CHARGE OF THE MODULE

HAMEURLAIN Abdelkader
Email : hameurlain@irit.fr

LEARNING GOALS

In the design and implementation framework of distributed database systems, the objective of the course is to present the main issues and introduce proposed methods.

SUMMARY OF THE CONTENT

1. Introduction to Distributed Databases
2. Main Functions of Distributed Database Systems
3. Software Architecture of Distributed Database Systems
4. Distributed Database Design
5. Principle of Distributed Query Processing

PREREQUISITES

Relational Database systems

REFERENCES

Principles of Distributed Database Systems, February 2011 M. Tamer Ozsu Patrick Valduriez Editor : Springer-Verlag New York Inc. ; Edition : 3rd, ed. 2011 (February 2011)

KEYWORDS

Distributed DB, Fragmentation, Localization, Distributed Query Optimization

UE	ANGLAIS	3 ECTS	2nd semester
EMINE2VM	TD : 24h		

TEACHER IN CHARGE OF THE MODULE

CHAPLIER Claire
Email : claire.chaplier@univ-tlse3.fr

UE	FRANÇAIS GRANDS DÉBUTANTS	3 ECTS	2nd semester
EMINE2YM	TD : 24h		

TEACHER IN CHARGE OF THE MODULE

JASANI Isabelle
Email : leena.jasani@wanadoo.fr Téléphone : 65.29

GLOSSARY

GENERAL TERMS

DEPARTMENT

The departments are teaching structures within components (or faculties). They group together teachers lecturing in one or more disciplines.

MODULE

A semester is structured into modules that may be mandatory, elective (when there is a choice) or optional (extra). A module corresponds to a coherent teaching unit whose sucessful completion leads to the award of ECTS credits.

ECTS: EUROPEAN CREDITS TRANSFER SYSTEM

The ECTS is a common unit of measure of undergraduate and postgraduate university courses within Europe, created in 1989. Each validated module is thus assigned a certain number of of ECTS (30 per teaching semester). The number of ECTS depends on the total workload (lectures, tutorials, practicals, etc.) including individual work. The ECTS system aims to facilitate student mobility as well as the recognition of degrees throughout Europe.

TERMS ASSOCIATED WITH DEGREES

Degrees have associated domains, disciplines and specialities.

DOMAIN

The domain corresponds to a set of degrees from the same scientific or professional field. Most of our degrees correspond to the domain Science, Technology and Health.

DISCIPLINE

The discipline corresponds to a branch of knowledge. Most of the time a discipline consists of several specialities.

SPECIALITY

The speciality constitutes a particular thematic orientation of a discipline chosen by a student and organised as a specific trajectory with specialised modules.

TERMS ASSOCIATED WITH TEACHING

LECTURES

Lectures given to a large group of students (for instance all students of the same year group) in lecture theatres. Apart from the presence of a large number of students, lectures are characterized by the fact they are given by a teacher who defines the structure and the teaching method. Although its content is the result of a collaboration between the teacher and the rest of the educational team, each lecture reflects the view of the teacher giving it.

TD : TUTORIALS

Tutorials are work sessions in smaller groups (from 25 to 40 students depending on the department) led by a teacher. They illustrate the lectures and allow students to explore the topics deeper.

TP : PRACTICALS

Teaching methods allowing the students to acquire hands-on experience concerning the knowledge learned during lectures and tutorials, achieved through experiments. Practical classes are composed of 16 to 20 students. Some practicals may be partially supervised or unsupervised. On the other hand, certain practicals, for safety reasons, need to be closely supervised (up to one teacher for four students).

PROJECT

A project involves putting into practice in an autonomous or semi-autonomous way knowledge acquired by the student at the university. It allows the verification of the acquisition of competences.

FIELD CLASS

Field classes are a supervised teaching method consisting of putting into practice knowledge acquired outside of the university.

INTERNSHIPS

Internships are opportunities enabling students to enrich their education with hands-on experience and to apply lessons learned in the classroom to professional settings, either in industry or in research laboratories. Internships are strongly regulated and the law requires, in particular, a formal internship convention established between the student, the hosting structure and the university.