Course: **English for professional and academic communication**

Semester: Fall, Spring

Credits: 6 ECTS

Contact: Francisca López / flopez@etsisi.upm.es

Objective/s:
The course is mainly practical and it is taught in English. It covers aspects related to the writing of professional and academic documents in English. Besides, it deals with oral communication abilities required for negotiation, meetings and professional presentations, among others.

Contents:

1. Terminology in academic environments and TICs.
2. General characteristics of written professional communication.
3. Types of professional communication: letters, e-mail, report, memorandum and other types of documents.
4. General characteristics of written academic communication.
5. Types of academic communication: abstracts, papers, Diploma projects and other types of documents.
6. Oral professional communication.
7. Facts intervening in oral presentations: context, partners, formality degree, body language and others.

Students must get a final grade of at least 5 out of 10 points to pass the course.

Course: **Frontiers in Computer Science**

Semester: Spring

ECTS: 6

Coordinator's name and e-mail: Victor Mitrana
/victor.mitrana@upm.es

Objective/s:

This course is focused on the fundamental models in Computer Science and their limitations. There will be presented three models of computation emerging from different ideas: standard programs, Turing machine, recursive functions. It will be shown that thay are equivalent supporting the Church-Turing thesis.
The course will discuss a few fundamental questions like:

- Are the von Neumann architecture and the Turing model sufficient for computing?
- Is anything beyond the traditional/classic computing paradigmes?
- Are other emerging paradigms including: quantum computing, optical computing, molecular computing, chemical computing, amorphous computing, etc. viable alternatives?

The didactic objectives are:

a) Learn a few of the fundamental models of comutation.
b) Learn their limitations and the reason of these limitations.
c) Find other computing paradigms and evaluate their viability.
d) Find the major drawbacks in the implementation of these new paradigms.

Contents:

1. Introduction to the theory of computation
2. Classic models of computation: standard programs, Turing machines, recursive functions.
3. Decidability and undecidability
4. Computational complexity basic concepts.
5. Non-convenional models of computation
6. State of the art in this area.

Assessment (brief description):

a) Continuous evaluation: 2*WE*0.30+ SA*0.20+WGP*0.20

   WE: Written exam; minimum 3 in 10
   SA: Seminar assignment; minimum 2 in 10
   WGP: Presentation of work performed in groups

b) Only Final exam and students that need July exam to pass the course: WE*1

WE: Written exam; minimum 5 in 10

Students must get at least 5 out of 10 points to pass the course.

Bibliography

1. M. Davis, R. Sigal, E.J. Weyuker:*Computability, Complexity, and Languages, Second Edition: Fundamentals of Theoretical Computer Science*, Morgan Kaufmann Publisher, 1994.
2. G. Rozenberg, T. Bäck, Joost N. Kok (Eds.) *Handbook of Natural Computing*, 4 volumes, Springer Verlag, 2012.

Course: **Information Coding**

Semester: Fall

Credits: 6 ECTS

Contact: Luis Miguel Pozo Coronado / luispozo@etsisi.upm.es

Objective/s:
The subject of this course is the study of the different possibilities to encode the information numerically, depending on the intended goal: conciseness (data compression), integrity (error detection codes) or security (cryptography).

The general objectives are: a) To understand the different mathematical concepts and tools underlying the models under consideration; and
b) To implement these models, with special attention to efficiency and security issues.

Contents:

1. Introduction to Information Coding: Lossless compression, Huffman codes. Error detection, CRC.
2. Introduction to Cryptology.
3. Computational Complexity.
4. Number theory.
5. Public key cryptosystems: RSA, ElGamal. Digital Signature.
6. Primality tests.

Assessment (brief description):

- Continuous evaluation:

Written exams: 40% (T1: 8%; T2-3: 12%; T4-6: 20%)
Moodle tests: 10%
Lab projects: 25% (4 projects, 5% each; Validation test: 5%)
Final project (Toolbox including all lab work): 25%

- Only Final exam and students that need July exam to pass the course:

Written exam: 50%
Lab exam (using the Toolbox procedures): 50%

Students must get at least 5 out of 10 points to pass the course.

Course: **Legal and professional issues**

Semester: Fall

Credits: 3 ECTS

Contact: Celia Fernández Aller / mariacelia.fernandez@upm.es

Objective/s:

This course is focused in the fundamentals and underlying legal concepts concerning Computer and Software Engineering: privacy, data protection, intellectual property rights, IT contracts and E-commerce.

The general objectives are: a) Learn the law and legal principles in the areas concerning IT Law. B) Recognize and apply the ethical and legal rules and standards of conduct involved in the practice of Engineering. C) Analyze legal problems and interactions between Law and IT Tecnologies.

Contents:

7. Introduction to IT Law
8. Data protection: Principles; rights and duties; liability; international transfers
9. Intellectual Property Rights: copyright, rules on fair use, special rules on copy protection for digital media; patents; trademarks; software licenses, end user license agreements, free software licenses and open-source licenses; professional liability of individual developers, warranties
10. IT contract: preparation, future-proofing the contract, negotiating liability and indemnities
11. E-commerce

Assessment (brief description):

- Continuous evaluation:

WE*0.30+ LA*0.30+WGP*0.40

WE: Written exam; minimum 3 in 10
LA: Lab assignment; minimum 3 in 10
WGP: Presentation of work performed in groups

- Only Final exam and students that need July exam to pass the course:

WE*1

WE: Written exam; minimum 5 in 10

Students must get at least 5 out of 10 points to pass the course.

Course: **Linux system administration**

Semester: Fall

Credits: 6 ECTS

Contact: Pilar Manzano / pilar.manzano@upm.es

Objective/s:
Understanding and doing some of the most important tasks of Linux system administration.

Contents:

1. Introduction
2. System startup and shutdown
3. Installing and updating SW
4. User accounts
5. Managing system resources
6. System security
7. Automating tasks
8. File systems and disks
9. Advanced use of disk
10. Backups
11. Printer management

Assessment (brief description):

- Continuous evaluation:

Partial exams: 20% (Test1: 10%; Test2: 10%)
Lab assignments: 20%
Generic competence (written and oral communication): 20%
Final exam: 40% (Students must get at least 5 out of 10 points to pass the course)

- Only Final exam and students that need July exam to pass the course:

Lab assignments: 20%
Generic competence (written and oral communication): 20%
Final exam: 60% (Students must get at least 5 out of 10 points in this exam to pass the course)

Students must get a final grade of at least 5 out of 10 points to pass the course.

Course: **Software Architecture and Design**
Semester: Spring
Credits: 6 ECTS

Contact: Agustín Yagüe (ayague@etsisi.upm.es)

Objective/s: This course is focused in the fundamentals of software architecture and design. In software engineering software architecture includes the process to define the software structure, components, modules, interfaces, test methods, and data required so that the system can satisfy the requirements. Alternatively, software architecture takes the requirements model to a more detailed model that represents the computing solution that, typically, will include architecture and detailed design specifications.

Contents: Software Architecture and lifecycle; architecture design; architecture views; architecture styles; architecture evaluation and refactoring; design in the lifecycle; design considerations; object oriented and non-object oriented design methods; design documentation; guidelines to select a design method; design assessment; verification and validation; architecture patterns; reference architectures; product lines.

Assessment (brief description):

- Continuous evaluation:

WE*0.25+ LA1*0.25+LA2*0.15+P*0.1+LLA1*0.5+ LLA2*0.5+WGP*0.15

WE: Written exam; minimum 3.5 over 10
LA: Lab assignment; minimum 3 over 10
LLA: Leadership in lab assignment
WGP: Presentation of work performed in groups

- Only final exam and students that need July exam to pass the course:

WE*0.4+ LA1*0.25+LA2*0.15+LLA1*0.5+ LLA2*0.5+IP*0.10

WE: Written exam; minimum 5 over 10
LA: Lab assignment
LLA: Leadership in lab assignment
IP: Presentation of work performed individually during the exam; minimum 4 over 10

Students must get at least 5 out of 10 points to pass the course

Course: **Software Quality**
Semester: Fall

Credits: 6 ECTS

Contact: Fernando Arroyo / farroyo@etsisi.upm.es

Objective/s:

- To learn and apply techniques for measuring software quality.
- To learn processes for quality management.

Contents:

1. Introduction
2. Quality techniques
3. Measuring software quality
4. Models and characteristics of software quality
5. Cost and ethics considerations
6. System quality requirements
7. Error characterization
8. Processes in quality management
9. Quality management regulation
10. Quality improvement
11. Tools for quality support

Students must get a final grade of at least 5 out of 10 points to pass the course.

Course: **Spanish for Science and Technology**

Semester: Fall, Spring

Credits: 3 ECTS

Information:

http://www.upm.es/Estudiantes/Movilidad/LenguasInternacionalizacion/AreaEspanol

---