



POLITÉCNICA

INTERNATIONAL
CAMPUS OF
EXCELLENCE

COORDINATION PROCESS OF
LEARNING ACTIVITIES
PR/CL/001



E.T.S. de Ingeniería de
Sistemas Informáticos

ANX-PR/CL/001-01

LEARNING GUIDE

SUBJECT

615000523 - Frontiers in computer science

DEGREE PROGRAMME

61IW - Grado en Ingeniería del Software

ACADEMIC YEAR & SEMESTER

2017/18 - Semester 2

Index

Learning guide

1. Description.....	1
2. Faculty.....	1
3. Prior knowledge recommended to take the subject.....	2
4. Skills and learning outcomes	2
5. Brief description of the subject and syllabus.....	4
6. Schedule.....	6
7. Activities and assessment criteria.....	8
8. Teaching resources.....	10

1. Description

1.1. Subject details

Name of the subject	615000523 - Frontiers in computer science
No of credits	6 ECTS
Type	Optional
Academic year of the programme	Third year
Semester of tuition	Semester 6
Tuition period	February-June
Tuition languages	English
Degree programme	61IW - Grado en Ingeniería del Software
Centre	Escuela Tecnica Superior de Ingeniería de Sistemas Informáticos
Academic year	2017-18

2. Faculty

2.1. Faculty members with subject teaching role

Name and surname	Office/Room	Email	Tutoring hours *
Fernando Arroyo Montoro	1111	fernando.arroyo@upm.es	Sin horario.
Victor Mitrana (Subject coordinator)	1227	victor.mitrana@upm.es	Sin horario. Tutoring shedule will be published in the Moodle of the Subject.

* The tutoring schedule is indicative and subject to possible changes. Please check tutoring times with the faculty member in charge.

3. Prior knowledge recommended to take the subject

3.1. Recommended (passed) subjects

- Estructura de datos
- Algorítmica y complejidad

3.2. Other recommended learning outcomes

El plan de estudios Grado en Ingeniería del Software no tiene definidos otros conocimientos previos para esta asignatura.

4. Skills and learning outcomes *

4.1. Skills to be learned

CT10 - Creatividad e innovación: Habilidad para presentar recursos, ideas y métodos novedosos y concretarlos en acciones. Capacidad para innovar en cada una de las obras. Resolver de forma nueva y original situaciones o problemas en el ámbito de la ingeniería.

CT12 - Uso de tecnologías de la información y las comunicaciones : Usar las tecnologías de la información y las comunicaciones en el ámbito de la ingeniería.

CT6 - Razonamiento crítico: La capacidad de pensar de manera crítica implica tres cosas: (1) una actitud de estar dispuesto a considerar de una manera reflexiva los problemas y asuntos que entran dentro del rango de las experiencias de uno, (2) conocimiento de los métodos de investigación lógica y el razonamiento, y (3) una cierta habilidad en la aplicación de esos métodos.

4.2. Learning outcomes

RA10 - Recopila y sintetiza información de fuentes bibliográficas y de clases magistrales en inglés

RA114 - Resuelve problemas abiertos, barajando varias alternativas, justificando las decisiones tomadas de manera razonada y crítica, expresando con precisión las argumentaciones necesarias y las conclusiones. Identifica situaciones en las que el planteamiento del problema exija una aproximación que no permita distintos niveles de abstracción o no sea posible una jerarquización de los mismos y su solución

RA117 - En un artículo científico-técnico, identifica el problema, los postulados y premisas del autor así como los conceptos necesarios para el razonamiento, distinguiendo los datos y las opiniones, presentando la información relevante de forma organizada y estableciendo conclusiones razonadas.

RA127 - Lee un texto un artículo científico de cierta complejidad, o maneja diferentes fuentes de información relativo a un tema no impartido previamente en clase, y distingue los conceptos principales de los secundarios, explica la relación entre dichos conceptos y sintetiza las ideas principales seleccionando la información relevante, realizando un resumen, mapa conceptual o esquema de mayor complejidad

RA263 - Escribe documentos con estructura compleja para documentar algún trabajo o estudio realizado. Organiza las distintas partes del texto siguiendo alguna estructura de texto científico: formulación de hipótesis, contexto, objetivos, justificación del trabajo, método utilizado, resultados obtenidos, conclusiones y referencias bibliográficas. Se expresa de manera eficaz mediante la escritura y los apoyos gráficos

RA119 - Presenta recursos, ideas y métodos novedosos y concretados en acciones. Resuelve de forma nueva y original situaciones o problemas en el ámbito de la ingeniería.

RA65 - Presenta recursos, ideas y métodos novedosos y concretados en acciones. Resuelve de forma nueva y original situaciones o problemas en el ámbito de la ingeniería

* The Learning Guides should reflect the Skills and Learning Outcomes in the same way as indicated in the Degree Verification Memory. For this reason, they have not been translated into English and appear in Spanish.

5. Brief description of the subject and syllabus

5.1. Brief description of the subject

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 they 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 paradigms?
- Are other emerging paradigms including: quantum computing, optical computing, molecular computing, chemical computing, amorphous computing, etc. viable alternatives?

The didactic objectives are:

1. Learn a few of the fundamental models of computation.
2. Learn their limitations and the reason of these limitations.
3. Find other computing paradigms and evaluate their viability.
4. Find the major drawbacks in the implementation of these new paradigms.

5.2. Syllabus

1. 1. Introduction to the theory of computation
 - 1.1. Basic concepts, what can be computed and the intrinsic limitations of computation.
 - 1.2. Deterministic and nondeterministic algorithms, semi-algorithms.
2. 2. Classic models of computation
 - 2.1. The standard language and standard programs. Godelization of programs.
 - 2.2. Turing machines: variants, relationships with standard programs.
 - 2.3. Recursive sets, recursively enumerable sets
3. 3. Decidability and undecidability
 - 3.1. What is a decision problem and how it can be formalized.
 - 3.2. Halting problem. Rice theorem and its applications. Post Correspondence Problem.
4. 4. Computational complexity basic concepts.
 - 4.1. Time complexity measure: computational model, classes, relationships.
 - 4.2. Space complexity measure: computational model, classes, relationships.
 - 4.3. NP-completeness : definition, methodology, examples.
5. 5. Non-conventional models of computation
 - 5.1. Non-classical algorithms: generating the space of all solutions, filtering
 - 5.1.1. Computational models inspired from nature.
6. 6. State of the art in this area
 - 6.1. Last results and experiments in the area of bio-inspired computing.

6. Schedule

6.1. Subject schedule*

Week	Face-to-face classroom activities	Face-to-face laboratory activities	Other face-to-face activities	Assessment activities
1	Lesson 1: Introduction to the theory of computation Duration: 02:00 Lecture Lesson 2: Classic models of computation Duration: 02:00 Lecture			
2	Lesson 2: Classic models of computation Duration: 02:00 Lecture		Practice of Lesson 2 Duration: 02:00 Problem-solving class	
3	Lesson 2: Classic models of computation Duration: 02:00 Lecture		Practice of Lesson 2 Duration: 02:00 Problem-solving class	
4	Lesson 3: Decidability and undecidability Duration: 02:00 Lecture		Practice of Lesson 2 Duration: 02:00 Problem-solving class	
5	Lesson 3: Decidability and undecidability Duration: 02:00 Lecture		Practice of Lesson 3 Duration: 02:00 Problem-solving class	
6	Lesson 3: Decidability and undecidability Duration: 02:00 Lecture		Practice of Lesson 3 Duration: 02:00 Problem-solving class	First exam (RA10, RA263, RA119, RA65, RA114) Written test Continuous assessment Duration: 02:00
7	Lesson 4: Computational complexity basic concepts Duration: 02:00 Lecture		Practice of Lesson 3 Duration: 02:00 Problem-solving class	
8	Lesson 4: Computational complexity basic concepts Duration: 02:00 Lecture		Practice of Lesson 4 Duration: 02:00 Problem-solving class	
9	Lesson 4: Computational complexity basic concepts Duration: 02:00 Lecture		Practice of Lesson 4 Duration: 02:00 Problem-solving class	
10	Lesson 5: Non-conventional models of computation Duration: 02:00 Lecture		Practice of Lesson 4 Duration: 02:00 Problem-solving class	

11	Lesson 5: Non-conventional models of computation Duration: 02:00 Lecture		Practice of Lesson 5 Duration: 02:00 Problem-solving class	
12	Lesson 6: State of the art Duration: 02:00 Lecture		Practice of Lesson 5 Duration: 02:00 Problem-solving class	
13	Seminar assignment Duration: 04:00 Cooperative activities			
14	Seminar assignment Duration: 04:00 Cooperative activities			Second exam (RA10, RA263, RA119, RA65, RA114) Written test Continuous assessment Duration: 02:00
15	Seminar assignment Duration: 04:00 Cooperative activities			Seminar assignment presentation (RA10, RA263, RA119, RA65, RA114, RA117, RA127) Group presentation Continuous assessment Duration: 04:00 Seminar assignment assessment (RA10, RA263, RA119, RA65, RA114, RA117, RA127) Other assessment Continuous assessment Duration: 00:00
16				Final exam (RA10, RA263, RA119, RA65, RA114, RA117, RA127) Written test Final examination Duration: 02:00
17				

The independent study hours are training activities during which students should spend time on individual study or individual assignments.

Depending on the programme study plan, total values will be calculated according to the ECTS credit unit as 26/27 hours of student face-to-face contact and independent study time.

* The subject schedule is based on a previous theoretical planning of the subject plan and might go through experience some unexpected changes along throughout the academic year.

7. Activities and assessment criteria

7.1. Assessment activities

7.1.1. Continuous assessment

Week	Description	Modality	Type	Duration	Weight	Minimum grade	Evaluated skills
6	First exam (RA10, RA263, RA119, RA65, RA114)	Written test	Face-to-face	02:00	30%	3 / 10	CT10 CT6
14	Second exam (RA10, RA263, RA119, RA65, RA114)	Written test	Face-to-face	02:00	30%	3 / 10	CT10 CT6
15	Seminar assignment presentation (RA10, RA263, RA119, RA65, RA114, RA117, RA127)	Group presentation	Face-to-face	04:00	20%	2 / 10	CT10 CT12 CT6
15	Seminar assignment assessment (RA10, RA263, RA119, RA65, RA114, RA117, RA127)	Other assessment	No Presential	00:00	20%	2 / 10	CT10 CT12 CT6

7.1.2. Final examination

Week	Description	Modality	Type	Duration	Weight	Minimum grade	Evaluated skills
16	Final exam (RA10, RA263, RA119, RA65, RA114, RA117, RA127)	Written test	Face-to-face	02:00	100%	5 / 10	CT10 CT12 CT6

7.1.3. Referred (re-sit) examination

Description	Modality	Type	Duration	Weight	Minimum grade	Evaluated skills
Final exam	Written test	Face-to-face	02:00	100%	5 / 10	

7.2. Assessment criteria

Assessment (brief description):

WE: Written exam; minimum 3 in 10

SA: Seminar assignment; minimum 2 in 10

WGP: Presentation of work performed in groups

Continuous evaluation: $2*WE*0.30+ SA*0.20+WGP*0.20$

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

Written exam, minimum 5 in 10

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

8. Teaching resources

8.1. Teaching resources for the subject

Name	Type	Notes
1. M. Davis, R. Sigal, E.J. Weyuker: Computability, Complexity, and Languages, Second Edition: Fundamentals of Theoretical Computer Science, Morgan Kaufmann Publisher, 1994.	Bibliography	
2. G. Rozenberg, T. Bäck, Joost N. Kok (Eds.) Handbook of Natural Computing, 4 volumes, Springer Verlag, 2012.	Bibliography	
M.R. Garey, D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W. H. Freeman & Co. Publisher, New York, NY, USA 1990.	Bibliography	
S. Arora, B. Barak, Computational Complexity: A Modern Approach, Cambridge University Press, 2009	Bibliography	
C.H. Papadimitriou, Computational complexity, John Wiley and Sons Ltd. Chichester, UK, 2003	Bibliography	