



Universitat de Lleida

DEGREE CURRICULUM  
**DISTRIBUTED COMPUTING AND  
APPLICATIONS**

Coordination: LERIDA MONSO, JOSEP LLUIS

Academic year 2021-22

## Subject's general information

<b>Subject name</b>	DISTRIBUTED COMPUTING AND APPLICATIONS			
<b>Code</b>	102027			
<b>Semester</b>	2nd Q(SEMESTER) CONTINUED EVALUATION			
<b>Typology</b>	Degree	Course	Character	Modality
	Bachelor's Degree in Computer Engineering	3	COMPULSORY	Attendance-based
	Bachelor's Degree in Computer Engineering	4	OPTIONAL	Attendance-based
<b>Course number of credits (ECTS)</b>	6			
<b>Type of activity, credits, and groups</b>	<b>Activity type</b>	PRALAB	TEORIA	
	<b>Number of credits</b>	3	3	
	<b>Number of groups</b>	1	1	
<b>Coordination</b>	LERIDA MONSO, JOSEP LLUIS			
<b>Department</b>	COMPUTER SCIENCE AND INDUSTRIAL ENGINEERING			
<b>Teaching load distribution between lectures and independent student work</b>	6 ECTS = 25x6 = 150 hours 40% -> 60 classroom hours 60% -> 90 hours of autonomous student work			
<b>Important information on data processing</b>	Consult <a href="#">this link</a> for more information.			
<b>Language</b>	The course will be teach 40% in Catalan and 60% in spanish.			

Teaching staff	E-mail addresses	Credits taught by teacher	Office and hour of attention
LERIDA MONSO, JOSEP LLUIS	josepluis.lerida@udl.cat	2	
SAPES PLANAS, JORDI		4	

## Subject's extra information

The course is eminently practical, so work practices and will have an important weight. Basically we work with two languages, C for parallel programming and Java for distributed objects (RMI). However, when we talk of distributed applications, as important as the program itself is the design of the application, which also intensively apply the concepts of software engineering.

To continue the course is essential for students to have good fundamentals and C programming Java. It is much harder, learn to develop distributed programs and parallel sequential programming if not previously mastered. In the course it is assumed that students are able to design, develop and debug sequential applications of medium difficulty without much trouble.

## Learning objectives

- Introduce the basic concepts on Distributed Computing and the organisation of distributed systems.
- Provide an overview of the main architectures of Distributed Computing and his impact on the information technologies.
- Assimilate the fundamental principles and the distinct types of underlying models of Distributed Computing.
- Know the main paradigms of Distributed Computing and understand his strong points, his disadvantages and main fields of application.
- Comprise the technological challenges that represent the utilisation, the design and the implementation of the distributed systems.
- Provide an overview of the distributed systems, analysing different cases of study and applying them to solve real problems in different fields of Distributed Computing.
- Develop the skills of design and analysis of distributed systems that help to comprise, evaluate the quality the solutions proposed
- Encourage the adoption of the distributed model for the resource sharing on large scale, in a transparent form and independently of his physical location.

## Competences

### Strategic Competences of the UdL:

- CT2. Mastering a foreign language, especially English.
- CT3. Training Experience in the use of the new technologies and the information and communication technologies.

### Cross-disciplinary competences:

- EPS11. Capacity to understand the needs of the user expressed in a no technical language.

### Specific Competences:

- T12. Capacity to choose, design, deploy, integrate, evaluate, build, manage, explode and keep the hardware, software and network technologies inside the cost and quality requirements.
- T15. Capacity to select, deploy, integrate and manage systems of information that satisfy the needs of the organisation, within the cost and quality requirements.
- T16. Capacity to conceive systems, applications and services based in network technologies, including Internet, web, e-commerce, multimedia, interactive services and mobile computation.

## Subject contents

### 1. Introduction to the Distributed Computing

1. Definitions and concepts
2. Distributed Computing Challenges
3. Types of Distributed Systems
  1. Distributed Computing systems
  2. Distributed Information systems
  3. Distributed Embedded systems
4. Distributed Systems Architectures

### 2. Paradigms of Distributed Computing

1. Message passing
2. Client-server
3. Peer-to-Peer
4. Messages systems
5. RPC (Remote Procedure Call)
6. Distributed Objects
7. Mobile agents
8. Component-based
9. Service-oriented

### 3. Distributed Objects

1. Introduction
2. Paradigm of distributed objects
3. RMI: Remote method invocation
4. Advanced RMI
  1. Callbacks
  2. Security Management in RMI

### 4. Hardware and Software of Parallel Computing

1. Parallel architectures
2. Networks of interconnection
3. Clusters
4. Job Queues

### 5. Parallel programming

1. The message passing model
2. The message passing interface
3. Basic routines
4. Point-to-point communications
5. Collective communications
6. Derived data-types
7. Routines for groups and communicators administration
8. Virtual topologies
9. MPI- 2 and MPI -3
  1. Cases of study

## Methodology

**Theory Sessions (3 credits)**

- Lecture: classes based on notes and transparencies where the concepts of the subject will be presented.
- Problems: The concepts of the subject will work through a series of exercises to be resolved collaboratively and help assimilate key concepts.
- Use Cases: It will apply the techniques seen in class to real examples and their impact on application performance will be analyzed.

**Laboratory Sessions (3 credits)**

- Tutorials and personalized monitoring by groups of practices.
- Laboratory: technologies and APIs for distributed programming will be presented and worked through tutorials and examples.
- Problems: Making and correcting exercises related to both the theoretical and practical part of the course.

**Autonomous work:**

- The homework exercises and practices will be completed outside of class time.
- Forums Tool. In this space the student can raise doubts regarding the contents seen in the Theory and Laboratory sessions as well as pose all kind of doubts about the project. All students are encouraged to participate in resolving the doubts of their peers. Teachers participate to clarify or resolve those doubts that have no answer from students.

**Development plan**

<b>Week</b>	<b>Description</b>	<b>Classroom Activity GG</b>	<b>Classroom Activity GM</b>	<b>Autonomous work activity</b>
1	Presentation Introduction	Subject presentation T1: Introduction of Distributed Computing	T1: Introduction of Distributed Computing	<b>Study literature and the program</b>
2	Introduction	T1: Introduction of Distributed Computing	T1: Introduction of Distributed Computing	Introduction
3	Introduction Paradigms	T1: Introduction of Distributed Computing	T2: Distributed Computing Paradigms	Paradigms tasks
4	Paradigms	T2: Distributed Computing Paradigms	T2: Distributed Computing Paradigms	Paradigms tasks
5	Paradigms	T2: Distributed Computing Paradigms	Presentation Introduction/ Paradigms tasks	Paradigms tasks
6	Distributed Objects	T3: Distributed Objects and Remote Invocation Practice 1: Presentation	T3: Distributed Objects and Remote Invocation <b>Tutorial: RMI</b>	Tutorial: Distributed Objects Practice 1
7	Distributed Objects	T3: Distributed Objects and Remote Invocation <b>Tutorial: RMI</b>	T3: Distributed Objects and Remote Invocation <b>Tutorial: RMI</b>	Tutorial: Distributed Objects Practice 1
8	Parallel Computing	T4: Parallel Computing Hardware & Software <b>Tutorial: SGE</b>	T4: Parallel Computing Hardware & Software <b>Tutorial: SGE</b>	Tutorial: SGE Practice 1
9		<b>1<sup>er</sup> Partial</b>		Study
10	Design of parallel applications	T4: Parallel Computing Hardware & Software <b>Tutorial: SGE</b>	Practice 1: Delivery	Tutorial: SGE
11	Design of parallel applications	T4: Parallel Computing Hardware & Software <b>Tutorial: SGE</b>	T5: Parallel programming <b>Tutorial: MPI</b>	Tutorial: SGE Tutorial MPI

12	Parallel programming	T5: Parallel programming <b>Tutorial: MPI</b> Practice 2: Presentation	T5: Parallel programming <b>Tutorial: MPI</b>	Tutorial MPI Practice 2
13	Parallel programming	T5: Parallel programming <b>Tutorial: MPI</b>	<b>Wednesday Holiday</b>	Tutorial MPI Practice 2
14	Parallel programming	T5: Parallel programming <b>Tutorial: MPI</b>	Problems: Implementing Parallel Programs	Tutorial MPI Practice 2
15	Parallel programming	Distributed Computing Projects	Practice 2: Delivery	Practice 2
16		<b>2<sup>nd</sup> Partial</b>		Study
17		<b>2<sup>nd</sup> Partial</b>		Study
18		TUTORIAS		
19		<b>Recovery</b>		Study

## Evaluation

Table. Assessment Activities

Acr.	Assessment activity	Weighting	Minimum Grade	In Group	Mandatory	Recoverable
P1	1st Partial Exam	15%	NO	NO	YES	YES
P2	2nd Partial Exam	15%	NO	NO	YES	YES
PRA	Practices	30%	4	NO	YES	YES
WRK	Works/Tutorials	35%	NO	YES/NO	NO	NO
PCL	Class participation	05%	NO	NO	NO	NO
<i>It must be approved all practices individually. A practice is considered suspended if fails to reach 4.</i>						
<b>FinalGrade</b> = 0.15*P1 + 0.15*P2 + 0.30*PRA + 0.35*WRK + 0.05*PCL						

The subject approves with a final mark upper or equal to 5 and having done properly the laboratory practices (all they with at least a mark of 4)

The final mark for the subject it is obtained from the pondered sum of the marks of the two examinations and the practices plus the class participation marks and the continued assessment.

The subject has two partial exams, each one of them with a weight of 15% in the final mark. These exams are mandatory.

The realisation and the overcoming of the laboratory practices is mandatory to approve the subject. The practices will be evaluated with a note that will represent 30% of the final note of the subject.

It does not exist minimum marks in the written exams to obtain the final mark of the subject. Only it exists a minimum note (4) for the laboratory practices.

## Bibliography

### Basic Bibliography:

- Coulouris G, Dollimore J., Kindberg T.: "Sistemas distribuidos: Conceptos y diseño"; Addison-Wesley, 2001.
- M.L. Liu, "Computación distribuida". Edt. Addison Wesley, 2004
- [Peter Pacheco](#), "An Introduction to Parallel Programming", Morgan Kaufmann, 2011.

### Extended Bibliography:

- M. Ben-Ari, "Principles of Concurrent and Distributed Programming", Addison-Wesley, 2nd Edition, 2006
- [Rajkumar Buyya](#): "High Performance Cluster Computing: Architectures and Systems", Edt. Pearson Education; 1st edition 1999
- [Rajkumar Buyya](#): "High Performance Cluster Computing: Programming and Applications", Volume 2 , Edt. Prentice Hall, 1st edition 1999.