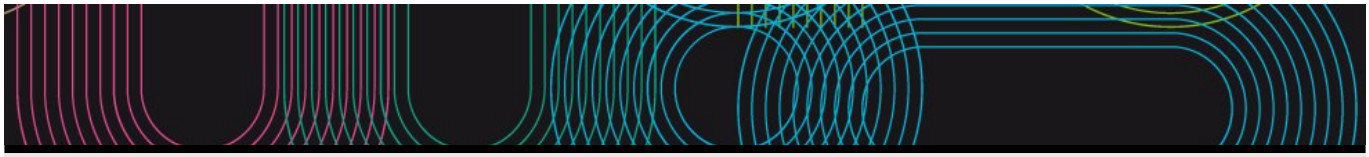

IDENTIFYING DATA				
Concurrency and distribution				
Subject	Concurrency and distribution			
Code	O06G151V01308			
Study programme	Grado en Ingeniería Informática			
Descriptors	ECTS Credits	Choose	Year	Quadmester
	6	Mandatory	3rd	2nd
Teaching language	#EnglishFriendly Spanish Galician			
Department				
Coordinator	Formella , Arno			
Lecturers	Formella , Arno			
E-mail	formella@uvigo.es			
Web	http://formella.webs.uvigo.es/doc/cdg22			
General description	The content forms the necessary basis to understand the operation of competing and / or distributed applications, the evaluation of competing algorithms, the description of data and information in distributed systems, the operation of modern processors, and the specific characteristics of the programming with processes / threads even in a distributed way. The classes are given mainly in Spanish. The student can choose to work in Galician, Spanish, German, Portuguese, and / or English. Certain additional information (such as manuals and additional information) will be given in English. English Friendly subject: International students may request from the teachers: a) materials and bibliographic references in English, b) tutoring sessions in English, c) exams and assessments in English.			

Skills	
Code	
A1	Students will have shown they have sufficient knowledge and understanding of an area of study, starting after completion of general secondary education, and normally reaching a level of proficiency that, being mostly based on advanced textbooks, will also include familiarity with some cutting-edge developments within the relevant field of study.
A2	Students will be able to apply their knowledge and skills in their professional practice or vocation and they will show they have the required expertise through the construction and discussion of arguments and the resolution of problems within the relevant area of study.
A3	Students will be able to gather and interpret relevant data (normally within their field of study) that will allow them to have a reflection-based considered opinion on important issues of social, scientific and ethical nature.
A5	Students will acquire the learning skills that are required to pursue further studies with a high degree of independence.
B5	Ability to conceive, develop and maintain computing systems, services and applications through use of software engineering methods as tools to ensure quality, according to the knowledge and training acquired.
B6	Ability to conceive and develop centralized or distributed computing systems and architectures, integrating hardware, software and networks, according to the knowledge and training acquired.
B9	Ability to solve problems by taking the initiative, making decisions and acting independently and creatively. Ability to communicate the knowledge contents, skills and abilities of the Computer Science Engineer profession.
C14	Ability to analyze, design, build and maintain applications in a robust, safe and efficient way, choosing the most appropriate paradigm and programming languages.
C15	Ability to know, understand and assess the structure and architecture of computers, as well as their basic components.
C16	Knowledge of the characteristics, functions and structure of Operating Systems and design and implementation of applications based on their services.
C17	Knowledge and application of the characteristics, functions and structure of Distributed Systems, Computer Networks and the Internet and design and implementation of applications based on them.

C20	Knowledge and application of the fundamental principles and basic techniques of parallel, concurrent, distributed and real-time programming.
C28	Ability to identify and analyze problems and design, develop, implement, verify and document software solutions on the basis of sound knowledge of the theories, models and techniques available nowadays.
D4	Analysis, synthesis and evaluation capacity
D5	Organizational and planning skills
D6	Ability to abstract: ability to create and use models that reflect real situations
D7	Ability to search, relate and structure information from various sources and to integrate ideas and knowledge.
D8	Ability to work in situations of lack of information and / or under pressure
D10	Interpersonal relationship skills.
D11	Critical thinking
D14	Have motivation for quality and continuous improvement

Learning outcomes

Expected results from this subject	Training and Learning Results			
RA1: To know the theoretical foundations of concurrent and distributed systems.	A1 A2 A3 A5	B5 B6 B9	C14 C15 C16 C17 C20 C28	D4 D5 D6 D7 D8 D11 D14
RA2: To know systems and environments with concurrency and distribution.	A1 A2 A3 A5	B5 B6 B9	C14 C15 C16 C17 C20 C28	D4 D5 D6 D7 D8 D11 D14
RA3: To know the process of generating applications for concurrent and distributed systems.	A1 A2 A3 A5	B5 B6 B9	C14 C15 C16 C17 C20 C28	D4 D5 D6 D7 D8 D10 D11 D14
RA4: To know the tools and their properties in use to generate code for concurrent and distributed systems.	A1 A2 A3 A5	B5 B6 B9	C14 C15 C16 C17 C20 C28	D4 D6 D7 D8 D10 D11 D14

Contents

Topic	
Concurrent and distributed systems	- Concept of concurrent and distributed programming - Introduction to the modeling of competing or distributed systems - Hardware architectures for the concurrence and distribution -Tools for the development of competing and distributed applications
Processes	- Concept of processes - Scheduler - Atomicity and mutual exclusion - Transactional concurrence - Clock and distributed status
Synchronisation and communication	- Synchronization and communication in concurrent and distributed systems - Synchronization and communication at the low level - Synchronization and communication at the high level - Security and vivacity in competing and distributed systems

Programming and application development tools	- Concurrent and distributed programming with JAVA - Concurrent and distributed programming with C / C ++ - Design patterns for the development of concurrent and distributed applications - Tools and methodologies of design, verification and debugging of competing and distributed applications

Planning			
	Class hours	Hours outside the classroom	Total hours
Introductory activities	0.5	0	0.5
Lecturing	18	10	28
Previous studies	0	18	18
Laboratory practical	24	26	50
Problem solving	1.5	19.5	21
Presentation	0	2	2
Discussion Forum	2.5	0	2.5
Problem and/or exercise solving	1	0	1
Essay questions exam	2	0	2
Report of practices, practicum and external practices	0	12	12
Laboratory practice	1	0	1
Problem and/or exercise solving	2	10	12
*The information in the planning table is for guidance only and does not take into account the heterogeneity of the students.			

Methodologies	
	Description
Introductory activities	Presentation of the subject and explanation of all the criteria of the syllabus. Time schedule of the face-to-face activities. Introduction of the tools of control and evaluation. Specific recommendations to achieve the aims of the subject.
Lecturing	Description of the theoretical contents of the course. Presentation of examples and cases studies. Previous readings. Control of knowledge acquisition by of the student. Interaction with/between the students via specific activities.
Previous studies	Reading of documents related with the contents of the course. Analysis and design of the tasks of the laboratory work.
Laboratory practical	Implementation and debugging of the exercises posed like programming tasks. Proofs of operation and/or performance of concurrent applications and distributed with a critical analysis of the observations.
Problem solving	Resolution of problems. Verification, correction and performance analysis. Implementation of alternative solutions. Critical analysis of the proposed solutions .
Presentation	Brief description of the milestones reached in the programming tasks and related exercises.
Discussion Forum	Discussion and debate in groups of specific topics and possible solutions to problems presented by the teachers.

Personalized assistance	
Methodologies	**Description**
Introductory activities	Presentation of the organisation, evaluation, and learning capacities.
Lecturing	The professor summarises the information and the knowledge inherent to the course contents, interrelating the different parts and linking the concepts between them, with the bibliography and with the practices.
Presentation	The student exposes to the teacher and/or to a group of students the design of a solution and the obtained results .
Laboratory practical	The student works in the tasks published during the course with the teachers' support.
Discussion Forum	The student asks questions that considers timely related with the course contents or the process of learning. The teacher tries to promote active debate between the participants.
Tests	**Description**
Problem and/or exercise solving	The student answers short written questions.

Essay questions exam	The student answers a set of questions with rational arguments.
Report of practices, practicum and external practices	The student elaborates reports documenting the decisions taken and the results obtained including critical reasoning.
Laboratory practice	The student demonstrates his/her implementations of the programming tasks according to the specified requirements.
Problem and/or exercise solving	The student provides an informal demonstrates that the solutions have the required properties.

Assessment

	Description	Qualification	Training and Learning Results
Presentation	(P5) Development of algorithms or applications and their analysis with a certain level of formalism to check the correctness and study the performance. Assessment with a score of 1-10, optional and voluntary participation. (RA1, RA2, RA3, RA4)	5	B5 C14 D4 B6 C15 D5 B9 C16 D6 C17 D7 C20 D8 C28 D11 D14
Problem and/or exercise solving	(P1) Set of short questions to control the performance of activities, tasks, and studies. Average of the tests carried out with a score of 1-10. (RA1, RA2, RA3, RA4)	10	A1 B5 C14 D4 A2 B6 C15 D5 B9 C16 D6 C17 D7 C20 D8 C28 D10 D11 D14
Essay questions exam	(P2) Set of long questions that relate the different sections of the content and measure the level of acquisition of the competences of the subject. Test with score of 1-10, minimum required: 4. (RA1, RA2, RA3, RA4)	40	A1 B5 C14 D4 A2 B6 C15 D6 B9 C16 D7 C17 D8 C20 D11 C28 D14
Report of practices, practicum and external practices	(P3) Preparation of reports (according to a guide) that collect the main developments and results obtained by the student. Some of these reports will be produced in small groups. Average of evaluations of the activities with scores of 1-10. (RA1, RA2, RA3, RA4)	25	A3 B5 C14 D4 A5 B6 C15 D5 B9 C16 D6 C17 D7 C20 D8 C28 D10 D11 D14
Laboratory practice	(P4) Demonstration of the developments and implementation of the programming tasks and study experiments. Average of evaluations of the activities with scores of 1-10., Minimum required: 4. (RA1, RA2, RA3, RA4)	25	A3 B5 C14 D4 A5 B6 C15 D5 B9 C16 D6 C17 D7 C20 D8 C28 D10 D11 D14

Other comments on the Evaluation

A student that does not participate in at least 80% of the face-to-face activities will be evaluated as "non-assistant". The decision of becoming "non-assistant" can be taken by the student or by the teacher (when it is no longer possible to reach 80% of assistance in the face-to-face activities).

Evaluation criteria for attendees of 1st edition:

Theory evaluation: completion of different activities (for example, reading bibliography, drawing diagrams, studying manuals, etc.) and verification through short written tests throughout the course (even unexpectedly) that will collect corresponding theoretical and practical content of the subject taught during face-to-face classes and in autonomous studies (P1). Taking a final written exam (P2) with different questions.

Practice evaluation: evaluates the completion of the proposed exercises (P4), the preparation of the documentation and related reports (P3). Completion of optional work towards the end of classes (P5). For attendees, the spirit of calculating the final grade for the subject is as follows: there is a combination of theoretical and practical tests throughout the course and, at the end, to assess the skills acquired. Good results in one part may compensate for not so good results in another part,

however a minimum should be achieved in the two most relevant sections (P2 and P4). The final grade is obtained as follows, assuming that each part (P1-P5) is evaluated on a scale of 0-10:

The course is failed if P2 grade less than 4 or P4 grade less than 4 (the grade of the part that motivates the fail will appear in the grade sheet). The course is approved if min (10, 0.1 * P1 + 0.4 * P2 + 0.25 * P3 + 0.25 * P4 + 0.05 * P5) greater than or equal to 5, if not it is failed.

Evaluation criteria for non-assistants: Methodology/Exam: the student is evaluated with a final written exam that covers all the content of the course

and evaluates the competencies acquired by the student (allowing a random sample to limit the time of the exam to 3 hours). % Rating: 95%. Evaluated competences: all. Evaluated learning outcomes: all. Methodology/Practices: Submission of the practices proposed during the laboratory hours. % Rating: 5%. Evaluated competences: all. Learning outcomes assessed: RA2, RA3, RA4.

Evaluation criteria for the 2nd edition:

Students who failed the first edition will have a similar assessment during the 2nd edition of evaluation. They have the possibility to redo any of the sections of the activities, except the sections "short questions" (P1) and "problem solving and / or recovery exercises" (P5). If they wish, these students can choose to be evaluated as non-assistant.

Evaluation criteria for the final degree edition:

Since the end-of-degree exams are carried out at the beginning of the course, it is evaluated without assistance with the possibility of evaluating the report and the deliveries of the practices through the work already carried out during the previous registered course.
The schedule of exams officially approved by the Board of the ESEI Center is published on the website http://www.esei.uvigo.es. There is coordination between the subjects of the same course to balance the continuous assessment process during the course.

Sources of information

Basic Bibliography

J.T. Palma Méndez, M.C. Garrido Carrera, F. Sánchez Figueroa, A. Quesada Arencibia., **Programacion Concurrente**, 84-9732-184-7, OUR 681.32 /392, Thomson, 2003

D. Lea, **Programación concurrente en Java**, 8478290389, OUR 681.32 /426, Addison Wesley, 2001

G. Coulouris, J. Dollimore, T. Kindberg, **Sistemas distribuidos : conceptos y diseño**, 84-7829-049-4, OUR 681.31 /27, Addison Wesley, 2001

M.L. Liu, **Computación distribuida : fundamentos y aplicaciones**, 8478290664, OUR 681.31 /201, Pearson/Addison Wesley, 2004

M. Herlihy, N. Shavit, **The Art of Multiprocessor Programming**, 9780123973375, OUR 681.32 /538, Morgan Kaufmann, 2012

C. Breshears, **The Art of Concurrency**, 9780596521530, OUR 681.32 /559, O'Reilly, 2009

Complementary Bibliography

D. Schmidt, M. Stal, H. Rohnert, F. Buschmann, **Pattern-oriented Software Architecture (v.2), Pattern for Concurrent and Networked Objects**, 978-0-471-48648-0, OUR 681.321 /16, John Wiley, 2007

Varios, **Internet**, http://www.java.com, Oracle, 2020

Varios, **Internet**, http://www.cppreference.com, 2020

Recommendations