

SUBJECT TEACHING GUIDE

G653 - Parallel, Concurrent and Real Time Programming

Degree in Computer Systems Engineering

Academic year 2024-2025

1. IDENTIFYING DATA					
Degree	Degree in Computer Systems Engineering			Type and Year	Compulsory. Year 3
Faculty	Faculty of Sciences				
Discipline	Subject Area: Computer Programming Compulsory Module				
Course unit title and code	G653 - Parallel, Concurrent and Real Time Programming				
Number of ECTS credits allocated	6	Term	Semester based (1)		
Web	http://www.istr.unican.es/assignaturas/ppctr/				
Language of instruction	Spanish	English Friendly	No	Mode of delivery	Face-to-face

Department	DPTO. INGENIERÍA INFORMÁTICA Y ELECTRÓNICA				
Name of lecturer	JOSE JAVIER GUTIERREZ GARCIA				
E-mail	josejavier.gutierrez@unican.es				
Office	Facultad de Ciencias. Planta: + 3. DESPACHO DE PROFESORES (3061)				
Other lecturers	ESTEBAN STAFFORD FERNANDEZ MARIO IBAÑEZ BOLADO				

3.1 LEARNING OUTCOMES

- To know the basics of computer programming and reasoning about programs in regard to parallel programming .

To know with familiarity the additional risks and difficulties of parallel programming and to be able to address them with the proper techniques.

To train students to conceive, specify, design, implement, and verify applications in which concurrent programming is used, providing the basics of concurrency, its advantages, problems and pathologies involved, as well as resources and synchronization primitives which historically have been introduced for safe concurrent programming.

To train students for the development of concurrent programs using programming languages that support concurrency or using the services of an operating system.

To train students to be able to design and analyze real-time uniprocessor systems, in which the temporal aspects are essential for the proper working of the application.

To enable students to develop simple real-time programs for uniprocessor by using programming languages or operating system services.

4. OBJECTIVES

To train students in the basic concepts of parallel programming . Concepts of parallel programming methodology that differ significantly from the methods used in the sequential and object-oriented programming will be introduced. A series of parallel algorithmic schemes, which are widely used in different applications will be presented . Referred to the programming model, we will focus on the paradigm of shared memory. For this purpose, programming practices will be developed using the OpenMP standard, as well as the C++ standard mechanisms.

To train students in the design, specification, implementation and verification of applications in which concurrent programming is used. In particular, the basic concepts of concurrency and synchronization, and formal methods for specifying and verifying concurrent programs will be introduced. We will also practice concurrent programming in Java and C on top of POSIX, identifying advantages and problems in contrast to sequential programming.

To train students in the design, analysis and implementation of simple real-time systems for uniprocessors. For this purpose, the different concepts that have been considered historically relevant for the proper scheduling of real-time systems will be introduced. Mechanisms for real-time programming with C/POSIX will be also put in practice.

6. SUBJECT PROGRAM	
CONTENTS	
1	<p>BLOCK I: Parallel Programming</p> <p>Chapter 1: Introduction to Parallel Programming Chapter 2: Parallelism in C++ Chapter 3: Performance analysis of parallel programs Chapter 4: Programming in Shared Memory: OpenMP Chapter 5: Optimizing the performance of OpenMP Programs</p> <p>OpenMP and C++ practices.</p>
2	<p>BLOCK II: CONCURRENT AND REAL-TIME PROGRAMMING</p> <p>1. Concurrency</p> <ul style="list-style-type: none"> - Introduction to concurrent programming - Synchronization - Concurrency and synchronization in programming languages and operating systems - Patterns and methods to express concurrency
3	<p>BLOCK II: CONCURRENT AND REAL-TIME PROGRAMMING</p> <p>2. Real Time</p> <ul style="list-style-type: none"> - Introduction to real-time systems: scheduling policies - Real-time system model: periodic events - Schedulability analysis - Synchronization protocols

7. ASSESSMENT METHODS AND CRITERIA				
Description	Type	Final Eval.	Reassessn	%
Parallel programming	Laboratory evaluation	No	Yes	33,33
Concurrent and real-time programming	Laboratory evaluation	No	Yes	66,67
TOTAL				100,00
Observations				
The final grade is obtained according to the following weights of the two thematic blocks: <ul style="list-style-type: none"> - Parallel programming 1/3 - Concurrent and real-time programming 2/3 One of the thematic blocks could be balanced with a minimum score of 4.0. In case one of the thematic blocks has a score lower than 4.0, the final grade will be the minimum of 4.9 and the average obtained. If only one of the thematic blocks is passed in the ordinary period, the grade of this part will be saved for the extraordinary period.				
Observations for part-time students				
Part-time students, who cannot follow the practices and continuous evaluation proposed, will be evaluated by equivalent tests to those established for the extraordinary period.				

8. BIBLIOGRAPHY AND TEACHING MATERIALS

BASIC

PROGRAMACIÓN PARALELA

- Francisco Almeida, Domingo Giménez, José Miguel Mantas, Antonio M. Vidal: "Introducción a la Programación Paralela". Editorial Paraninfo. 2008

PROGRAMACIÓN CONCURRENTE Y DE TIEMPO REAL

- Andy Wellings: "Concurrent and Real-Time Programming in Java". Wiley, 2004.

- J.S.W.Liu. "Real Time Systems". Prentice Hall, 2000.

- JAVADOC, referencia de la Api J2SE 8.0. Disponible online.<http://docs.oracle.com/javase/8/docs/api/>