Course guide
230490 - CPIA - Computer Programming and Its Applications

Last modified: 25/05/2023

Unit in charge:	Barcelona School of Telecommunications Engineering
Teaching unit:	701 - DAC - Department of Computer Architecture.
Degree:	BACHELOR'S DEGREE IN ENGINEERING PHYSICS (Syllabus 2011). (Optional subject).

Academic year: 2023 **ECTS Credits:** 6.0 **Languages:** Spanish

LECTURER

Coordinating lecturer:	Consultar aquí / See here: https://telecos.upc.edu/ca/estudis/curs-actual/professorat-responsables-coordinadors/responsables-assignatura
Others:	Consultar aquí / See here: https://telecos.upc.edu/ca/estudis/curs-actual/professorat-responsables-coordinadors/professorat-assignat-idioma

PRIOR SKILLS

None

TEACHING METHODOLOGY

The theoretical contents of the subject are taught in theory classes. These classes are complemented with practical examples and problems that students must solve in the Autonomous Learning hours.
In the laboratory sessions, the knowledge acquired in the theory classes is consolidated by solving programming problems related to the theoretical contents. During the laboratory classes, the teacher will introduce new techniques and will leave an important part of the class for the students to work on the proposed exercises.

Teaching methodology:
Application Classes
Expository classes
Laboratory classes
Group work (non face-to-face)
Individual work (not face-to-face)
Laboratory practice
Project

LEARNING OBJECTIVES OF THE SUBJECT

Demonstrate knowledge and understanding of the internal workings of a computer.
Analyze, design, build and maintain applications in a robust, safe and efficient way, choosing the most appropriate paradigm and programming languages.
Design, write, test, debug, document, and maintain code in a high-level language to solve programming problems by applying algorithmic schemes and using data structures.
Demonstrate knowledge and ability to apply the fundamental principles and basic techniques of sequential, parallel and concurrent programming.
Apply simple optimizations to code snippets to improve their performance across the architecture.

STUDY LOAD

Type	Hours	Percentage
Hours large group	26,0	17.33
Hours small group	39,0	26.00
Self study	85,0	56.67

Total learning time: 150 h

CONTENTS

Computer Architecture

Description:
Abstraction levels: Physical, Operation system and application Processor Architecture: functional units, registers, control unit, microprogramming; processing unit; pipelining. Microprocessors evolution; performance metrics. Processor and memory acceleration techniques. Memory hierarchy. Introduction to Storage and I/O Overview of multicore systems, multiprocessors and clusters: parallel processing, classification, modern many-core high-performance computing architecture(GPU) and multiprocessor systems networks.

Full-or-part-time: 9h
Theory classes: 2h
Practical classes: 1h 30m
Laboratory classes: 1h 30m
Self study : 4h

C Programming Language

Description:
Data types, operators and operations. Sentences (assignment, conditional and iterative constructors). Functions. Parameters by reference and value. Strings. Basic data structures. Arrays, structs. Recursion. Files. Examples.

Full-or-part-time: 26h
Theory classes: 4h
Practical classes: 3h
Laboratory classes: 3h
Self study : 16h

Python Programming Language

Description:
Operations and data types. Algorithmic constructs (sequences, conditionals and iterations statements). Strings, list, tuples, maps, sets and dictionaries. Functions. Classes and methods in Python. Recursion. Modules specifics: SciPy, NumPy, Matplotlib, Pyro, PuLP, Numba, Panda, others. Examples.

Full-or-part-time: 35h
Theory classes: 6h
Practical classes: 4h 30m
Laboratory classes: 4h 30m
Self study : 20h

Parallelism: Programming models for shared memory architectures

Description:
Parallelism and concurrence. Processes and threads. Performance metrics, speedup, scalability, Amdahl’s law. Introduction to OpenMP. Abstractions for parallel programming: task parallelism, data parallelism. Synchronization. Load balancing. Concurrency models for C and Python. Examples.

Full-or-part-time: 80h
Theory classes: 14h
Practical classes: 10h 30m
Laboratory classes: 10h 30m
Self study : 45h

GRADING SYSTEM

Periodic lab-made deliverables that require implementation using C or Python programming languages (L).

There will also be a final project where a code in pairs will be developed in Python/C and report or short video (PF) will be prepared.

The grade course will be obtained from the following formula: 0.25*L+0.75*PF

EXAMINATION RULES.

Evaluation standards: It is mandatory to attend 80% of the corresponding practices and deliver the questionnaires within the established classes.

BIBLIOGRAPHY

Basic:
- Grama, A.; Karypis, G.; Kumar, V.; Gupta, A. Introduction to parallel computing. 2nd ed. Harlow, England: Pearson Education, 2003. ISBN 9780201648652.
- Jiменéz, M.; Otero, B. Fundamentos de Ordenadores: programación en C [on line]. Barcelona: Iiciativa Digital Politècnica, 2013 [Consultation: 19/07/2022]. Available on: https://upcommons.upc.edu/handle/2099.3/36593. ISBN 9788476539958.
- Lutz, M. Learning python [on line]. 5th ed. Sebastopol, CA: O'Reilly, 2013 [Consultation: 07/07/2022]. Available on: https://ebookcentral-proquest-com.recursos.biblioteca.upc.edu/lib/upcatalunya-ebooks/detail.action?pq-origsite=primo&docID=1619476. ISBN 9781449355722.
- Zaccone, G. Python parallel programming cookbook. Birmingham, UK: Packt Publishing, 2015. ISBN 9781785289583.
- OpenMP application programming interface: version 5.0 [on line]. OpenMP, 2018 [Consultation: 28/09/2023]. Available on: https://www.openmp.org/wp-content/uploads/OpenMP-API-Specification-5.0.pdf.

Complementary:
- Culler, D.E.; Singh, J.P.; Gupta, A. Parallel computer architecture: a hardware/software approach. San Francisco: Morgan Kaufmann Publishers, 1999. ISBN 9781558603431.
- Stallings, W. Organización y arquitectura de computadores [on line]. 7a ed. Madrid: Prentice Hall, 2006 [Consultation: 19/07/2022]. Available on: https://www-ingebook-com.recursos.biblioteca.upc.edu/ib/NPcd/IB_BooksVis?cod_primaria=1000187&codigo_libro=1266. ISBN 9788489660823.
- Hennessy, J.L.; Patterson, D.A. Computer architecture: a quantitative approach [on line]. Cambridge, MA: Elsevier, Morgan Kaufmann, 2019 [Consultation: 19/07/2022]. Available on: https://ebookcentral-proquest-com.recursos.biblioteca.upc.edu/lib/upcatalunya-ebooks/detail.action?pq-origsite=primo&docID=787253. ISBN 9780128119051.