Course guide
230490 - CPIA - Computer Programming and Its Applications

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: 2022 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

<table>
<thead>
<tr>
<th>Type</th>
<th>Hours</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hours small group</td>
<td>39,0</td>
<td>26.00</td>
</tr>
<tr>
<td>Self study</td>
<td>85,0</td>
<td>56.67</td>
</tr>
<tr>
<td>Hours large group</td>
<td>26,0</td>
<td>17.33</td>
</tr>
</tbody>
</table>

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:

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

Python Programming Language

Description:

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:

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:

Complementary: