Course guides
240132 - 240132 - Informatics

Unit in charge: Barcelona School of Industrial Engineering
Teaching unit: 723 - CS - Department of Computer Science.
Degree: BACHELOR'S DEGREE IN INDUSTRIAL TECHNOLOGY ENGINEERING (Syllabus 2010). (Compulsory subject).
Academic year: 2020 ECTS Credits: 4.5 Languages: Catalan

LECTURER

Coordinating lecturer: Tost Pardell, Daniela
Others: Dolors Ayala, Antoni Soto, Daniela Tost, Marc Vigo, Josep Vilaplana.

DEGREE COMPETENCES TO WHICH THE SUBJECT CONTRIBUTES

Specific:
1. Basic knowledge on the use and programming of computers, operative systems, data bases and computer software with an engineering application.

TEACHING METHODOLOGY

The three hours a week of attended classes for the course are divided into one-hour lecture in large groups, and two hours of lab class in small groups. Given the extraordinary conditions of the semester, the theory sessions will be given on-line with videos and other materials elaborated by the teaching staff. There will only be a 1 hour weekly class, remote and live for all groups, on Tuesdays from 7pm to 8pm, coinciding with the theoretical schedule of group 40 and 50.

LEARNING OBJECTIVES OF THE SUBJECT

1. This subject proposes an approach to computing in width. Assuming the basics of programming, focuses on design issues and, using modules, introduces key issues using the computer programming language chosen as a binder.
2. Acquire the ability to analyze problems of some complexity, and applying techniques of structured analysis and design, performed with skill algorithms correct, legible, efficient and easy to maintain.
3. Knowing how to structure information and evaluate the various possible representations in a computer environment.
4. Being able to use abstract models for solving real problems.
5. Design of numerical applications.
6. Make a medium scale computer project.

STUDY LOAD

<table>
<thead>
<tr>
<th>Type</th>
<th>Hours</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hours large group</td>
<td>15,0</td>
<td>13.33</td>
</tr>
<tr>
<td>Hours medium group</td>
<td>30,0</td>
<td>26.67</td>
</tr>
<tr>
<td>Self study</td>
<td>67,5</td>
<td>60.00</td>
</tr>
</tbody>
</table>

Total learning time: 112.5 h
CONTENTS

**Topic 1. Object-oriented design**

**Description:**
Introduction and use of object-oriented design.

1. Inheritance, polymorphism.
2. Abstract data types.
3. Basic Design Patterns.

**Related competencies:**
CE3. Basic knowledge on the use and programming of computers, operative systems, data bases and computer software with an engineering application.

**Full-or-part-time:** 26h 30m
Theory classes: 4h
Laboratory classes: 8h
Self study: 14h 30m

**Topic 2. Recursivity**

**Description:**
Introduction and use of the recursive design.

Recursive design, completion and correctness.

**Related competencies:**
CE3. Basic knowledge on the use and programming of computers, operative systems, data bases and computer software with an engineering application.

**Full-or-part-time:** 24h
Theory classes: 3h
Laboratory classes: 6h
Self study: 15h

**Topic 3. Data Structures**

**Description:**
Introduction and use of some data structures.

1. We present some data structures and some application examples of them.
2. Data structures to be presented are stacks, lists, queues, trees or graphs.

**Related competencies:**
CE3. Basic knowledge on the use and programming of computers, operative systems, data bases and computer software with an engineering application.

**Full-or-part-time:** 20h
Theory classes: 3h
Laboratory classes: 6h
Self study: 11h
### Topic 4. Iterators

**Description:**
Programming with iterators and iterables

- Generator functions
- Iterable classes
- Introduction to functional programming through iterators

**Related competencies:**
CE3. Basic knowledge on the use and programming of computers, operative systems, data bases and computer software with an engineering application.

**Full-or-part-time:** 20h
- Theory classes: 2h
- Laboratory classes: 4h
- Self study: 14h

---

### Topic 5. Utility Modules

**Description:**
Introduction and utilization of some existing modules in Python.

1. Operating system services.
2. Mathematical software.
3. Services and web applications.
4. Documentation and test programs.

**Related competencies:**
CE3. Basic knowledge on the use and programming of computers, operative systems, data bases and computer software with an engineering application.

**Full-or-part-time:** 20h
- Theory classes: 3h
- Laboratory classes: 6h
- Self study: 11h
GRADING SYSTEM

Throughout the course, 5 exercises will be proposed to be solved at home (Li, i = 1..5) and 4 exercises to be solved and delivered in face-to-face lab sessions (Ei, i = 1..4). The grade of the subject (NF) is calculated from the grades of these assignments (AC), the theory grade (ET) and the final exam grade (EF) as follows:

\[ NF = \max (AC, 0.2 \times AC + 0.2 \times ET + 0.4 \times EF) \]

where

\[ AC = \text{Continuous evaluation} = 0.4 \times LNP + 0.6 \times LP \]

\[ LNP = \text{Non-presential deliverables} = \frac{\text{suma (Li)}}{5} \text{ amb } i = 1 \ldots 5 \]

\[ LP = \text{Presential deliverables} = \frac{\text{suma (Ei)}}{4} \text{ amb } i = 1 \ldots 4 \]

The LNP grade will be the arithmetic mean of the grades of the deliverable problems.

and the final exam grade (EF) as follows:

\[ NF = \max (AC; 0.4 \times AC + 0.2 \times ET + 0.4 \times EF) \]

where

\[ AC = \text{Continuous assessment grade} = 0.4 \times LNP + 0.6 \times LP \]

\[ LNP = \text{Non-classroom deliverables grade} = \frac{\text{sum (Li)}}{5} \text{ amb } i = 1 \ldots 5 \]

\[ LP = \text{Classroom deliverables grade} = \frac{\text{sum (Ei)}}{4} \text{ amb } i = 1 \ldots 4 \]

If, due to health circumstances, only one of the face-to-face sessions could be carried out, the final grade would be as:

\[ NF = \text{Final grade} = 0.4 \times LNP + 0.3 \times LP + 0.1 \times ET + 0.2 \times EF \]

where LP is the mean of the deliverables that could be done presentially.

If, due to health circumstances, none of the face-to-face sessions could be carried out, the final grade would be calculated as:

\[ NF = 0.5 \times LNP + 0.2 \times ET + 0.3 \times EF \]

In all cases, the revaluation grade will replace grade A.

BIBLIOGRAPHY

Basic:

Complementary:

RESOURCES

Hyperlink:
- Diposit obert de material de l'assignatura. Resource

Other resources:
- http://wiki.python.org/moin/UsefulModules
- http://www.python.org/community/sigs/current/edu-sig/
- L'introduction to Programming using Python], http://www.pasteur.fr/formation/infobio/python