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.

Learning objectives of the subject

1. This subject proposes an approach to computing in width. Assuming assumed 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>Total learning time: 112h 30m</th>
<th>Hours large group: 15h</th>
<th>13.33%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hours medium group: 30h</td>
<td></td>
<td>26.67%</td>
</tr>
<tr>
<td>Hours small group: 0h</td>
<td></td>
<td>0.00%</td>
</tr>
<tr>
<td>Guided activities: 0h</td>
<td></td>
<td>0.00%</td>
</tr>
<tr>
<td>Self study: 67h 30m</td>
<td></td>
<td>60.00%</td>
</tr>
</tbody>
</table>
# Content

## Topic 1. Object-oriented design

**Learning time:** 26h 30m  
Theory classes: 4h  
Laboratory classes: 8h  
Self study: 14h 30m

**Description:**  
Introduction and use of object-oriented design.  
1. Inheritance, polymorphism.  
2. Abstract data types.  
3. Basic Design Patterns.

## Topic 2. Recursivity

**Learning time:** 24h  
Theory classes: 3h  
Laboratory classes: 6h  
Self study: 15h

**Description:**  
Introduction and use of the recursive design.  
Recursive design, completion and correctness.

## Topic 3. Data Structures

**Learning time:** 20h  
Theory classes: 3h  
Laboratory classes: 6h  
Self study: 11h

**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.
Topic 4. Iterators

Description:
Programming with iterators and iterables

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

Learning 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.

Learning time: 20h
- Theory classes: 3h
- Laboratory classes: 6h
- Self study: 11h

Qualification system

The grade for the course (A) is calculated from the partial exam grade (P), the theoretical grade (T) and the final exam grade (F) as follows:

\[ A = 0.2 \cdot T + \max(0.2 \cdot P + 0.6 \cdot F, 0.8 \cdot F) \]

https://translate.google.com
Bibliography

Basic:


Complementary:


Others resources:

- http://wiki.python.org/moin/UsefulModules
- http://www.python.org/community/sigs/current/edu-sig/

Hyperlink

Dipòsit obert de material de l'assignatura