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.

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>Total learning time: 112h 30m</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hours large group:</td>
</tr>
<tr>
<td>15h</td>
</tr>
<tr>
<td>Hours medium group:</td>
</tr>
<tr>
<td>30h</td>
</tr>
<tr>
<td>Hours small group:</td>
</tr>
<tr>
<td>0h</td>
</tr>
<tr>
<td>Guided activities:</td>
</tr>
<tr>
<td>0h</td>
</tr>
<tr>
<td>Self study:</td>
</tr>
<tr>
<td>67h 30m</td>
</tr>
<tr>
<td>Topic 1. Object-oriented design</td>
</tr>
<tr>
<td>--------------------------------</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

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

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

<table>
<thead>
<tr>
<th>Topic 2. Recursivity</th>
<th>Learning time: 24h</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Theory classes: 3h</td>
</tr>
<tr>
<td></td>
<td>Laboratory classes: 6h</td>
</tr>
<tr>
<td></td>
<td>Self study: 15h</td>
</tr>
</tbody>
</table>

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

Recursive design, completion and correctness.

<table>
<thead>
<tr>
<th>Topic 3. Data Structures</th>
<th>Learning time: 20h</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Theory classes: 3h</td>
</tr>
<tr>
<td></td>
<td>Laboratory classes: 6h</td>
</tr>
<tr>
<td></td>
<td>Self study: 11h</td>
</tr>
</tbody>
</table>

**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.
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

**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:

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

Basic:


Complementary:


Others resources:

- http://wiki.python.org/moin/UsefulModules
- http://www.python.org/community/sigs/current/edu-sig/
- ¿Introduction to Programming using Python¿, http://www.pasteur.fr/formation/infobio/python

Hyperlink

Dipòsit obert de material de l'assignatura