Programming

Academic Year: (2023 / 2024) **Review date: 26/04/2023 15:21:34**

Department assigned to the subject: Computer Science and Engineering Department

Coordinating teacher: IGLESIAS MAQUEDA, ANA MARIA

Type: Basic Core ECTS Credits : 6.0

Year : 1 Semester : 1

Branch of knowledge: Engineering and Architecture

REQUIREMENTS (SUBJECTS THAT ARE ASSUMED TO BE KNOWN)

No pre-requisites

LEARNING OUTCOMES

CB1. Students have demonstrated possession and understanding of knowledge in an area of study that builds on the foundation of general secondary education, and is usually at a level that, while relying on advanced textbooks, also includes some aspects that involve knowledge from the cutting edge of their field of study.
CB2. Students are able to apply their knowledge to their work or vocation in a professional manner and possess the competences usually demonstrated through the development and defence of arguments and problem solving within their field of study.
COCIN3. Knowledge of basic and technological subject areas that will capacitate them to acquire new methods and theories and endow them with the versatility to adapt to new situations.
COCIN4. Ability to resolve problems with initiative, decision-making, creativity, and critical reasoning skills and to communicate and transmit knowledge, skills and abilities in the Industrial Engineering field.
CEP2. Knowledge and ability to apply computational and experimental tools for analysis and quantification of electrical engineering problems.
CEB3. Basic knowledge of the use of computer programming, operating systems, databases, and computer programs with engineering applications.

By the end of this content area, students will be able to have:
RA1.1. Knowledge and understanding of the programming foundations and computer systems underlying their branch of engineering.
RA1.4. Awareness of the wider multidisciplinary context of engineering.
RA2.1. The ability to apply their knowledge and understanding to identify, formulate and solve engineering problems using computer-aided methods.
RA5.2. The ability to combine theory and practice to solve engineering problems using computer-aided methods.

OBJECTIVES

By the end of this subject, students will be able to have:
1. Knowledge and understanding of the programming foundations and computer systems underlying their branch of engineering.
2. Awareness of the wider multidisciplinary context of engineering.
3. The ability to apply their knowledge and understanding to identify, formulate and solve engineering problems using computer methods.
4. The ability to combine theory and practice to solve engineering problems using computer methods.

DESCRIPTION OF CONTENTS: PROGRAMME

DESCRIPTION OF CONTENTS:
The purpose of the course is to give students an overview on programming techniques. As programming

language, it will be used a imperative programming language. The language used during this academic course is Python

PROGRAMME:

1. Programming foundations
Description: This chapter introduces the essential components of computer programming and programming languages.
Detailed contents:
- Basic architecture of computers
- Computer programming
- Programming paradigms
- Types of programming languages

2. Design of programs
Description: This chapter focuses on the internal design of programs, paying special attention to the concept of algorithm.
Detailed contents:
- Computer algorithms
- Analysis of algorithms
- Data structures

3. Coding
Description: Acquiring knowledge on coding by using an imperative programming language.
Detailed contents:
- Program data
- Operators
- Advanced data structures
- Program statements
- Subprograms

4. Testing and debugging
Description: Learning principles and techniques about testing, debugging and deploying computer programs.
Detailed contents:
- Compilation-execution cycle
- Testing techniques
- Debugging techniques

LEARNING ACTIVITIES AND METHODOLOGY

1) Theoretical lectures: 1,5 ECTS. Página 1 de 2Lectures oriented to present the theoretical concepts on programming.
2) Practical lectures: 1,5 ECTS. Classes in computer labs oriented to learn the use of an IDE and put into practice the syntax.
3) Programming exercises: 2,0 ECTS. Problem-based learning. Programming different pieces of code with the purpose of understanding those conceptual,
technical, and methodological principles that underlie structured programming.
4) Individual study: 1,0 ECTS. Self-studying to prepare for partials and final exams

ASSESSMENT SYSTEM

% end-of-term-examination/test:	30
% of continuous assessment (assigments, laboratory, practicals…):	70

- Midterm exam on programming foundations: 10%
- Programming projects: 60%
- End-of-term exam: 30%

There is a minimum mark required on the final exam of 5.0 of 10.0

BASIC BIBLIOGRAPHY

- Peter Wentworth, Jeffrey Elkner, Allen B. Downey and Chris Meyers How to Think Like a Computer Scientist: Learning with Python 3, https://media.readthedocs.org/pdf/howtothink/latest/howtothink.pdf , 2018

- Ravi Sethi. Programming Languages. Concepts and Constructs., ADDISON-WESLEY..

- Stephenson, Ben. The Python Workbook, Springer, 2014

ADDITIONAL BIBLIOGRAPHY

- Allen B. Downey Think Python, O'Reilly Media, Inc., 2012

- Bill Lubanovic Introducing Python, O'Reilly Media, Inc., 2014

- George W. Gorsline. Computer Organization: Hardware Software., PRENTICE HALL INTERNATIONAL EDITIONS..

- González Duque, R. Python para todos, http://mundogeek.net/tutorial-python/.

- Guido van Rossum and the Python Development Team Python Tutorial Rel. 3.7.0. (tutorial oficial de Python), https://docs.python.org/3/tutorial/ , 2017

- Stephen D.Burd. System Architecture. Hardware and Software in Business Information Systems., BOYD AND FRASER PUBLISHINGCOMPANY..