

Programming

Academic Year: (2024 / 2025)

Review date: 29-04-2024

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 : 2

Branch of knowledge: Engineering and Architecture

REQUIREMENTS (SUBJECTS THAT ARE ASSUMED TO BE KNOWN)

-Experience in the use of computers will be valuable.

SKILLS AND 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.

CB3. Students have the ability to gather and interpret relevant data (usually within their field of study) in order to make judgements which include reflection on relevant social, scientific or ethical issues.

CB4. Students should be able to communicate information, ideas, problems and solutions to both specialist and non-specialist audiences.

CB5. Students will have developed the learning skills necessary to undertake further study with a high degree of autonomy.

CG2. Learn new methods and technologies from basic scientific and technical knowledge, and being able to adapt to new situations.

CG3. Solve problems with initiative, decision making, creativity, and communicate and transmit knowledge, skills and abilities, understanding the ethical, social and professional responsibility of the engineering activity. Capacity for leadership, innovation and entrepreneurial spirit.

CG5. Use the theoretical and practical knowledge acquired in the definition, approach and resolution of problems in the framework of the exercise of their profession.

CE3. Use and program computers, operating systems, databases and software with application in engineering, and implement numerical algorithms in low and high level languages.

CT1. Work in multidisciplinary and international teams as well as organize and plan work making the right decisions based on available information, gathering and interpreting relevant data to make judgments and critical thinking within the area of study.

RA1. To have acquired sufficient knowledge and proved a sufficiently deep comprehension of the basic principles, both theoretical and practical, and methodology of the more important fields in science and technology as to be able to work successfully in them.

RA2. To be able, using arguments, strategies and procedures developed by themselves, to apply their knowledge and abilities to the successful solution of complex technological problems that require creating and innovative thinking.

RA3. To be able to search for, collect and interpret relevant information and data to back up their conclusions including, whenever needed, the consideration of any social, scientific and ethical aspects relevant in their field of study.

RA6. To be aware of their own shortcomings and formative needs in their field of specialty, and to be able to plan and organize their own training with a high degree of independence.

DESCRIPTION OF CONTENTS: PROGRAMME

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

Theoretical Lectures: 1 ECTS

Practical Lectures: 1 ECTS

- Exercise resolution
- Best coding practices
- General tutoring

Team Work: 3 ECTS

- Project design and development
- Application of best coding practices

Individual Work: 1 ECTS

- Contribution to team project
- Study and preparation of exams

ASSESSMENT SYSTEM

% end-of-term-examination:	30
% of continuous assessment (assignments, laboratory, practicals...):	70

CONTINUOUS EVALUATION (70%)

- Questionnaires of continuous evaluation: 10%
- Final project: 60% (with partial deliveries to follow up the status)

% end-of-term-examination:	30
% of continuous assessment (assignments, laboratory, practicals...):	70

FINAL EVALUATION (30%)

- Final exam: 30%

A minimum grade of 5.0 both in the final project and in the final examination is required to pass the course.

BASIC BIBLIOGRAPHY

- Allen B. Downey Think Python: How to Think Like a Computer Scientist, 2nd edition, O'Reilly, 2015
- Andrew Koenig C Traps and Pitfalls, Addison-Wesley Professional, 1989
- Anthony Scopatz, Kathryn D. Huff Effective Computation in Physics: Field Guide to Research with Python, O'Reilly, 2015
- Brian W. Kernighan / Dennis Ritchie The C Programming Language, Pearson, 2015
- David M. Beazley Python Cookbook: Recipes for Mastering Python 3, O'Reilly, 2011
- Jose María Álvarez Rodríguez Hands on Programming with Python: Theory and Practice, Amazon KDP Publishing, 2020
- K. N. King C Programming: A Modern Approach, 2nd Edition, W. W. Norton & Company, 2008
- Luciano Ramalho Fluent Python, O'Reilly, 2015
- Mark Lutz Learning Python, O'Reilly, 2013
- Paul Barry Head-First Python, 2nd edition, O'Reilly, 2016
- Robert C. Martin Clean Code: A Handbook of Agile Software Craftsmanship , Prentice Hall, 2008
- Samuel P. Harbison, Guy L. Steele Jr. C: A Reference Manual, 5th Edition, Pearson, 2002
- Zed A. Shaw Learn Python 3 the Hard Way, Addison-Wesley, 2016

ADDITIONAL BIBLIOGRAPHY

- Peter Prinz, Tony Crawford C in a Nutshell, O'Reilly Media, 2015
- Richard M. Reese Understanding and Using C Pointers, O'Reilly Media, 2013

BASIC ELECTRONIC RESOURCES

- Jose María Álvarez Rodríguez . Hands on Programming with Python: <https://chemaar.github.io/python-programming-course/>
- Python Community . Real Python Tutorials: <https://realpython.com/>

- Python Software Foundation . Python documentation and official resources: <https://www.python.org/doc/>

- Qingkai Kong, Timmy Siauw, Alexandre Bayen . Python Programming And Numerical Methods: A Guide For Engineers And Scientists: <https://pythonnumericalmethods.berkeley.edu/notebooks/Index.html>

- The Python Software Foundation . The Python Tutorial: <https://docs.python.org/3/tutorial/>