Programming

Academic Year: (2016 / 2017) **Review date: 29/04/2016 14:09:01**

Department assigned to the subject: Computer Science and Engineering Department

Coordinating teacher: IGLESIAS MARTINEZ, JOSE ANTONIO

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)

(None)

OBJECTIVES

Competencias genéricas:

- Synthesis and analysis
- Organization and planning
- Application of theoretical knowledge to practical problems
- Use of computers
- Proactivity, creativity and reasoning for problem resolution
- Team work

Specific Competences:

- Cognitive:
  - o To understand the role of Computer Science and Programming in the context of Industrial Engineering
  - o To solve engineering problems by designing algorithms and developing computer programs
  - o To understand the fundamentals of structured and modular programming
  - o To apply theoretical knowledge to solve practical problems by implementing computer programs in the C programming language
  - o To understand the role of Computer Science and Programming in the context of Industrial Engineering

- Procedural:
  - o To use a personal computer
  - o To solve engineering problems by designing and implementing computer programs in the C language
  - o To acquire a good programming style, resulting in efficient, well-organized, and well-documented programs
  - o To acquire the ability to understand and use third-party programs
  - o To learn how to use an integrated development environment (IDE) for program coding, compiling, and debugging
  - o To learn how to use common C programming libraries.

DESCRIPTION OF CONTENTS: PROGRAMME

Topic 1. Introduction to computer science and programming.
- Computer science and computers. A historical perspective.
- Information representation in computers
- Algorithms and programs. Tools for algorithm design

Topic 2. Software and Hardware

- Programs and programming languages. Classification. Translators, compilers and interpreters. Operating systems.
- Functional structure of a computer. Internal architecture. Instruction execution by the processor. Data storage. Peripherals. Computer networks and the Internet

Topic 3. Basic elements of the C programming language.
- General structure of a program
- Variables and constants
- Types of operators: arithmetic, relational, logic and assignment operators.
- Operators, expressions and instructions
- Pointer type.
- Input and output instructions

Topic 4. Control Flow and Loops
- Selection structures: if-else, switch
- Repetition structures (loops): for, while, do-while
- Nested control structures

Topic 5. Functions
- Modular programming
- Function definition
- Calling a function
- Types of arguments: input, output, input / output
- Passing Arguments by Value or by Reference -Scope of Function Variables. Visibility
- Arrays and structures as parameters
- Library functions and standard C libraries

Topic 6. Complex Data Types
- Introduction: structured vs simple data types
- Definition and use of arrays
- Pointers and arrays
- Character strings
- User defined data structures: records
- Arrays of records

Topic 7. Search, sort and merge algorithms
- Search algorithms
- Sort algorithms
- Merge algorithms

Topic 8. Advanced Topics
- External data structures: files and databases
- Dynamic memory allocation
- Computer programs commonly used in engineering.

LEARNING ACTIVITIES AND METHODOLOGY

Theory classes:
Basic theoretical knowledge and skills will be presented in large groups.

Resolution of Exercises:
Resolution of exercises by the student that will serve as self-evaluation and to acquire the procedural and cognitive competences.

Laboratory sessions:
Small groups classes, in which problems proposed to the students are discussed and developed using the computer.

Tutorials.

Final Practice.

ASSESSMENT SYSTEM

% end-of-term-examination/test:	50
% of continuous assessment (assigments, laboratory, practicals…):	50

Continuous Evaluation Exams:

Weeks 8 and 11 (approximately) - Date to be confirmed 15 days before the exam.
Content: First Exam: Units 3 and 4. Second Exam: Units 5, 6 and 7.
One single problem, writing a C program

Value: 20% of the final course mark (10% each exam).

Practical assignment (final project):

To be delivered in the last week of the course. There will be also four (evolving) previous deliveries related with some parts the final project. In addition, a practical final exam needs to be done.
The final project consists of a problem of higher complexity as compared to those solved during the course. It will be solved in teams of two students.
The mark will come out from:
* Mark of the four deliveries: 16% (4 deliveries * 4%/delivery = 16%).
* Mark of the final project delivery: 4%.
* Mark of the Exam related to the final project: 10%

Total value: 30% of the final course mark

Final exam:

Value: 50% of the final course mark
Two parts:
* Test, covering all topics of the course. Multiple choice test, four answers, only one correct. Penalty for wrong answers is one third. 30% of the exam mark.
* Two problems, where the student will be asked to design a C program. Each problem will account for 35% of the exam mark.

If the mark in the end-of-term exam is below 4.0, the student will not pass the course.

Extraordinary Exam:

The exam will have the same structure as the regular exam. The mark assigned to the continuous evaluation will be taken into account if it improves the final mark.

BASIC BIBLIOGRAPHY

- Al Kelley and Ira Pohl A book on C : programming in C , Addison-Wesley, 1998

- Brian W. Kernighan, Dennis M. Ritchie C Programming Language, Prentice Hall, 1988

- Deitel, Harvey M. C : how to program, Prentice-Hall International, 1994

- K. N. King C Programming: A Modern Approach, W.W. Norton & Company, 2008

- Paul J. Deitel, Harvey M. Deitel C: How to Program, Prentice Hall, 2009

- Paul J. Deitel, Harvey M. Deitel C: How to Program, Prentice Hall, 2009