Universidad Zaragoza

Academic Year/course: 2024/25

69168 - Fundamentals of Computing for Robotics, Graphics and Computer Vision

Syllabus Information

Academic year: 2024/25
Subject: 69168 - Fundamentals of Computing for Robotics, Graphics and Computer Vision
Faculty / School: 110 - Escuela de Ingeniería y Arquitectura
Degree: 615 - Máster Universitario en Robótica, Gráficos y Visión por Computador / Robotics, Graphics and Computer Vision
ECTS: 3.0
Year: 1
Semester: First semester
Subject type: Optional
Module:

1. General information

The course targets to foster effective computer and software engineering practices, withing the context of Robotics, Graphics, and Computer Vision systems. During the course the students, will learn the principles of computer system design and how to efficiently develop algorithms targeting them, avoiding the pitfalls provided by the lack of methodology in the development. The techniques explored in this course range from assembly and unmanaged memory, to high level abstractions such as functional or object oriented programming. Additionally, general software development concepts, such as test-driven development and version control will be also explored.

This course is recommended for students with some background in programming that want to improve their skills to confidently tackle the computing tasks during the rest of the Master studies.

2. Learning results

The student must be able to:

- Understand the architecture of computing systems (hardware and software)
- Design and implement simple algorithms to solve simple problems.
- Compile programs and link libraries against programs.
- Understand System Interfaces: Application Programming Interface, Application Binary Interface, Supervisor Binary Interface
- Understand memory management and how it is accessed when developing algorithms.
- Improve the robustness of algorithms through debugging and safe development methodologies.
- Abstract functionality and data representation in algorithms.
- Apply collaborative methodologies for software development.
- Understand the basics of Computer Security

3. Syllabus

- Review of imperative programming: sequential, conditional and iterative composition
- Subprograms: functions, parameters, variables and functional abstraction
- Review of Computing Systems (compute and networks): hardware and peripherals, operating systems, libraries and runtimes, applications
- Memory management: virtual memory, unmanned vs managed, reserving memory (stack and heap), indirect memory access (pointers, references), smart pointers
- Data abstraction: object oriented programming, inheritance, generic programming
- Robustness, debugging and testing: manual debugging methodologies, debuggers, exceptions, test-driven development
- Development practices and tools: IDEs, linterns, memory checkers, version control
- Basic security: data at rest, data in movement and data in compute

4. Academic activities

The course consists of 3 ECTS credits that correspond to an estimated student dedication of 75 hours distributed as follows:

- Theoretical classes: The theoretical concepts of the subject will be explained and illustrative practical examples will be developed to support the theory when necessary. (10h)
- Practical classes: Problems and practical cases will be carried out as a complement to the theoretical concepts studied. (5h)

- Laboratory practices: There will be a series of guided work tutored by the teacher. (15h)
- Study and assimilation of the theory exposed in the master classes. (30h)
- Practical application or research work (13h).
- Assessment tests (2h).

5. Assessment system

Continuous evaluation: for continuous evaluation, the assessment system is:

- **Practical assignments (80%):** A set of guided practices will be carried out throughout the course, with fixed deadlines to deliver each one. The outcome of these will be source code that will be evaluated according to its quality w.r.t. the principles, techniques and procedures presented during the theoretical classes.
- **Oral exams (20%):** With each practical assigment, the students will be asked questions about their work and how it is linked with the concepts presented during the theoretical classes.

Global evaluation: for global evaluation, the assessment system is:

- **Practical exam (80%):** Consisting on a set of practical programming exercises, this exam will be carried out on the official exam date. The solutions to this exam will be will be evaluated according to their quality w.r.t. the principles, techniques and procedures presented during the theoretical classes.
- **Oral exam (20%):** After the programming exercises, the students will be asked questions from the teachers regarding their proposed solutions and how are they linked with the concepts presented during the theoretical classes.

6. Sustainable Development Goals

8 - Decent Work and Economic Growth
9 - Industry, Innovation and Infrastructure