COURSE DETAILS

"TECNOLOGIE INFORMATICHE PER L'AUTOMAZIONE INDUSTRIALE"

SSD ING-INF/04

DEGREE PROGRAMME: BACHELOR DEGREE IN COMPUTER ENGINEERING

ACADEMIC YEAR: 2023-2024

GENERAL INFORMATION – TEACHER REFERENCES

TEACHER: GIANMARIA DE TOMMASI
PHONE: 081 768 3853
EMAIL: DETOMMAS@UNINA.IT

GENERAL INFORMATION ABOUT THE COURSE

INTEGRATED COURSE (IF APPLICABLE): N.A.
MODULE (IF APPLICABLE): N.A.
CHANNEL (IF APPLICABLE): N.A.
YEAR OF THE DEGREE PROGRAMME (I, II, III): III
SEMESTER (I, II): II
CFU: 6

REQUIRED PRELIMINARY COURSES (IF MENTIONED IN THE COURSE STRUCTURE "REGOLAMENTO")

Programmazione.

PREREQUISITES (IF APPLICABLE)

Basic knowledge of closed-loop control systems; basic knowledge of issues related to determinism in the design and development of real-time software systems.

LEARNING GOALS

The aim of the course is to educate the student to the problems of software design of industrial automation systems. The experimentation of the salient phases of the design and software implementation of automation systems through the use of professional tools and plant simulators is foreseen. At the end of the course the student will know the principles of operation and programming of control devices and their main requirements.

EXPECTED LEARNING OUTCOMES (DUBLIN DESCRIPTORS)

Knowledge and understanding

The training course aims to provide students with the methodological tools for the design of automation software based on Programmable Logic Controllers (PLCs). The main requirements for industrial control devices will be introduced, to then focus on the operating principle and programming languages for PLCs. The student must demonstrate that he has learned what are the peculiar requirements of hardware and software systems dedicated to the control of industrial processes. The student will also demonstrate the knowledge of the main phases of design of an automation system and the role of validation of control logics using simulation tools.

Applying knowledge and understanding

The student must demonstrate to be able to formalize through formal languages (such as, for example, the Sequential Functional Chart, SFC) the closed-loop operating specifications expressed in natural language for simple processes to be automated. Starting from the formal specifications, then, the student must demonstrate the ability to develop simple automation algorithms and implement them on a PLC using the languages provided by the IEC 61131-3 standard. The student will have to show the ability to develop simple synoptics to be used as a *user panel* for the management of an industrial plant. Finally, the student will have to show the ability to design the validation tests of the control logic also using simple plant simulators.

Communication skills

During the course there are classroom exercises in which students (in groups of 3/5 people) will have to develop and validate through the use of simulators, simple automation software. Through this activity, the student will have the opportunity to develop soft skills related to group work. In addition, these exercises will require to develop part of the solution outside of class hours, so the student will have the opportunity to have to report briefly to the teacher about the work done.

Learning skills

For the practical part of the course we will refer to the CODESYS tool (https://www.codesys.com/). However, since this tool is *fully compliant* **with the IEC 61131-3** standard, the student will not be limited in any way by the tool adopted and will learn the concepts related to IEC 61131-3 that are independent of the particular implementation. Therefore the student will acquire the ability to use in a short time any commercial development environment for PLC

.

COURSE CONTENT/SYLLABUS

- Control devices
  - Requirements for a control device
  - Controllers for general applications
  - Specialized controllers
- Programming of control devices
  - The programmable logic controller (PLC)
  - The IEC 61131-3 standard
    - Variables and variable types
    - Programming languages
      - Structured Text
      - Ladder Diagram
      - Functional Block Diagram
      - Instruction List
    - Program organization units (POUs)
      - Functions and Function Blocks
      - Programs
      - Tasks
      - Resources
      - Configuration
    - Sequential Functional Chart (SFC)
  - Supervision, control and data acquisition systems (SCADA)
  - Development cycle of automation systems
  - Validation of simple automation logics through the use of plant simulators (digital twin)

READINGS/BIBLIOGRAPHY

Chiacchio e F. Basile, “*Tecnologie Informatiche per l'Automazione*”, seconda ed., McGraw-Hill, 2004.

TEACHING METHODS

The teacher will use: a) lectures for about 60% of the total hours, b) classroom exercises through the use of the CODESYS tool (https://www.codesys.com/) for about 40% of the total hours.

EXAMINATION/EVALUATION CRITERIA

a) Exam type:

Exam type	
written and oral	X
only written	
only oral	
project discussion	
other	

In case of a written exam, questions refer to:	**Multiple choice answers**	
	Open answers	X
	Numerical exercises	

The written test is aimed at verifying the student's ability to design simple automation algorithms and consists in the solution of 1 or 2 simple industrial automation problems, a solution that requires development of control algorithms in one or more of the graphical programming languages provided by the standard IEC 61131-3 (Ladder Diagram, Function Block Diagram, Sequential Functional Chart). Typically the student has 3 hours for the written test.

The oral interview follows the written test and is aimed at a critical discussion of the solution (s) given by the student to the problems proposed in the written test, and to ascertain the acquisition of concepts and content introduced during the lessons.

b) Evaluation pattern: The outcome of the written test is binding for access to the oral test. The written and oral tests each contribute to 50% of the final evaluation, therefore passing the written test is not sufficient to pass the exam.