1CC1000 – Information Systems and Programming

Instructors: Gianluca Quercini
Department: DÉPARTEMENT INFORMATIQUE
Language of instruction: ANGLAIS, FRANCAIS
Campus: CAMPUS DE PARIS - SACLAY, CAMPUS DE RENNES, CAMPUS DE METZ
Workload (HEE): 60
On-site hours (HPE): 36,00

Description

This course consists of two parts : information systems and programming.

The aim is to introduce prospective engineers to the information systems that they will use throughout their careers and to enable them to properly design and write a computer program.

First year students have varying levels in these fields, depending on their previous experiences (some might already have participated in programming competitions) and education paths.

This course does not require any advanced notion in algorithmic.
At the end of the course all students will achieve a minimum common level of preparation; advanced students with previous knowledge in the subject will be able to improve their skills.

Three successive courses of the engineering program contribute to training students in programming and algorithms. This course is the first, followed by the Coding Weeks (two weeks dedicated to the design and implementation of a software project) and Algorithms and Complexity.

Quarter number

SG1

Prerequisites (in terms of CS courses)

None in terms of CentraleSupélec courses, SIP being the first computer science course of the engineering program.

Before the course, students will have to take a test to determine their level in Python : beginner, intermediate or advanced.
Students will be assigned to lab groups based on their initial level in Python.

The result of the test does not have any impact in the final grade of the course.

Syllabus

The course consists of 24 slots of 1.5 hours, with 6 lectures ("amphis"), 17 tutorials ("TD") and a final exam (1h30).

Information Systems :
- Data modelling and relational databases.
- SQL.
- Introduction to computer architecture.
- Network and security basics.

Programming & Python
- Interpreter, command line.
- Variables, objects types.
- Tests, loops, lists, sets, functions, modules, dictionaries, files, databases.
- Finding and using external libraries (numpy, matplotlib)
- Processing unstructured data (text).
- Debugging, test plan, « clean code ».
- GIT.

Class components (lecture, labs, etc.)

The course proposes a series of tutorials ("TD") with classic pencil-and-paper exercises or using a computer.
Students must install on their personal computer a programming environment (Visual Studio Code + Python + Git) before the course.

A technical support is provided by DISI (the CentraleSupélec technical service) to help all students with the installation process.

Classrooms are equipped with a wired or wireless Internet access, which allows students to look up resources on the Internet.
Students are supposed to do their exercises by themselves, although discussions among classmates and the professor is encouraged.

Grading

A written exam is organised at the end of the course. The exam is 1.5 hours long and it covers the entire course (information systems and programming). Students are not allowed to use their computers and the course material, with the only exception of a Python reference sheet.

A continuous assessment is also guaranteed with 3-4 short MCQ that the students will take during the course.
The MCQ are not mandatory, but students are strongly encouraged to take them in order to assess their progression.

Course support, bibliography

The course support is :
- A course book on Information Systems (in English).
- A reference manual for Python and "clean code" (in English).

Resources

Teaching staff (instructor(s) names): Guillaume Mainbourg, Dominique Marcadet, Marc-Antoine Weisser (lectures in French), Gianluca Quercini (lectures in English).

• Maximum enrollment (default 35 students): 30-35 students for each tutorial class ("TD").
• Required hardware : a personal computer with a minimal configuration (communicated to the students before the beginning of the course).
• Software, number of licenses required: open-source software, or free for students.
• Equipment-specific classrooms (specify the department and room capacity) : no

Learning outcomes covered on the course

Information Systems:

- Understand how computers and networks work.

- Know how to model and use a relational database.

- Understand the basics of security of information systems.

Programming:

- Write a program in a high level programming language to solve a problem.

- Apply good programming practices : modularity, code reuse, naming conventions, comments.

- Test a software : debugging, assertions, unit tests.

- Use software development tools: integrated development environment, command-line interface, Git.

Description of the skills acquired at the end of the course

• C2.1 Deepen a field of engineering sciences or a scientific discipline
• C6.1 Solve a problem numerically.
• C6.2 Design software.
• C6.3 Process data.