

## Programming 2

1. **Course number and name:** 020IF2NI3/020PR2NI3 Programming 2
2. **Credits and contact hours:** 4 ECTS credits, 2x1:15 contact hours + 3x2:30 overall lab sessions
3. **Name of course coordinator:** Chantal Saad Hajjar
4. **Instructional materials:** course handouts; lab experiments; slides; in-class problems
5. **Specific course information**
  - a. **Catalog description:**

This course allows the students to acquire advanced concepts of structured programming in Python. It also covers the basic concepts of object-oriented programming and their application to data abstraction and encapsulation by introducing the concepts of object instantiation, member visibility, inheritance, and polymorphism. Students will also learn how to create an ergonomic standalone graphical user interface using the standard *tkinter* library.
  - b. **Prerequisites:** 020IF1NI2/020PR1NI2 Programming 1
  - c. **Required/Selected Elective/Open Elective:** Required
6. **Educational objectives for the course**
  - a. **Specific outcomes of instruction:**
    - Create/use/update basic Python data structures in order to represent data models in basic applications.
    - Generate a Python list using list comprehension.
    - Employ error handling in programs.
    - Use classes for data abstraction.
    - Explain and apply class constructors.
    - Modify the states of class attributes.
    - Distinguish the roles of private members from public ones.
    - Apply operator overloading.
    - Recognize inheritance from a defined design of classes.
    - Apply inheritance when needed.
    - Employ polymorphism.
    - Recognize the specific role of different widgets in graphical user interfaces (GUIs).
    - Label some events in GUIs.
    - Write code for event handling in GUIs.
    - Sketch a simple GUI for a given problem.
    - Write code for simple GUI programs.

**b. PI addressed by the course:**

<b>PI</b>	<b>1.2</b>	<b>1.3</b>
<b>Covered</b>	x	x
<b>Assessed</b>	x	x

**7. Brief list of topics to be covered**

- Strengthening the concepts of structured programming in Python (4 lectures)
- List Comprehension (1 Lecture)
- Dictionaries (2 lectures)
- Sets (1 lecture)
- Advanced topics on variables and functions (2 lectures)
- Error handling (1 lecture)
- Introduction to object-oriented programming, classes, objects, attributes, constructors, methods (2 lectures)
- Class attributes, class methods, static methods (1 lecture)
- Encapsulation, operator overloading (3 lectures)
- Inheritance & polymorphism (3 lectures)
- Introduction to Graphical User Interfaces (GUI), standard widgets (2 lectures)
- Advanced GUI: menus and multiple windows (2 lectures)