# Object-Oriented Programming

1. **Course number and name:** 020CPPES1 Object-Oriented Programming

2. **Credits and contact hours:** 6 ECTS credits, 2x1:15 course hours + 2:30 lab hours

3. **Instructor's or course coordinator's name:** Dany Mezher

4. **Text book:**
   Thinking in C++, 2nd Edition
   a. **Other supplemental materials:**
      Professor textbook and course material, E-learning support using Moodle, MOOC on coursera.org

5. **Specific course information**
   a. **Catalog description:**
      Program structure, types, literals and variables, operators, program control instructions (conditions and loops), functions, arrays, structures - Object-Oriented Programming: Objects and classes, attributes and methods, constructor and destructor, encapsulation, inheritance, virtual functions, abstract classes et polymorphism, method and operator overloading, exceptions, Input/Output, streams, generics and templates, Standard Templates Library (STL), Graphical User Interfaces with Qt.

   b. **Prerequisites or co-requisites:**

   c. **Required:** Required for CCE and EE students

6. **Specific goals for the course**
   a. **Specific outcomes of instruction:**
      - Write complex programs in C++
      - Maintain existing C++ programs

   b. **KPI:**

| KPI | c1 | i2 | k1 | k2 | k3 |
|---|---|---|---|---|---|
| Covered | x | x | x | x | x |
| Assessed | | | x | x | x |

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

| #Lectures | Topic | Lecture | Lab |
|---|---|---|---|
| 1 | Introduction, Compiled vs Interpreted languages | 1 | |
| 2 | C++ program structure, expressions and instructions, data types, declarations and operators | 2 | |

| | | | |
|---|---|---|---|
| 2 | Lab 1. Introduction to the Integrated Development Environment (IDE). Visual studio | | 2 |
| 1 | Program Flow control: Conditional instructions, loops, break, continue | 1 | |
| 2 | Lab 2. Using the debugger to locate program bugs | | 2 |
| 2 | Functions, passing arguments, return values | 2 | |
| 1 | Advanced data types: long, signed and unsigned modifiers, arrays (1D, 2D…), structures types | 1 | |
| 2 | Lab 3. Arrays, loops and conditions | | 2 |
| 2 | Pointers | 2 | |
| 2 | Lab 4. Functions & pointers | | 2 |
| 2 | Objects: Abstraction, classes, attributes, methods, encapsulation | 2 | |
| 2 | Lab 5. Objects (encapsulation) | | 2 |
| 1 | constructors: Constructors, default constructor, conversion constructor, copy constructor…. Destructors | 1 | |
| 2 | Lab 6. Object construction, destruction | | 2 |
| 2 | Inheritance, static attributes, static methods | 2 | |
| 2 | Lab 7. Inheritance | | 2 |
| 3 | Polymorphism: virtual methods, pure virtual methods, static vs dynamic name resolution, abstract classes | 3 | |
| 2 | Lab 8. Polymorphism | | 2 |
| 2 | Operator overloading | 2 | |
| 2 | Lab 9. Operators | | 2 |
| 1 | Exceptions | 1 | |
| 2 | Lab 10. Exceptions | | 2 |
| 3 | Input/output | 3 | |
| 2 | Lab 11. Read and writing to files | | 2 |
| 1 | Type casting, namespaces and templates | 1 | |
| 2 | Lab 12. | | 2 |
| 4 | GUI with Qt | 4 | |
| 2 | Lab 13. | | 2 |
| 2 | Lab 14. | | 2 |