# COURSE OUTLINE

## (1) GENERAL

| | | | | |
|---|---|---|---|---|
| **SCHOOL** | Engineering | | | |
| **DEPARTMENT** | Electrical and Computer Engineering | | | |
| **LEVEL OF STUDY** | Undergraduate | | | |
| **COURSE UNIT CODE** | 1.006 | **SEMESTER OF STUDY** | $1^{st}$ | |
| **COURSE TITLE** | Scientific Programming with Python | | | |
| **COURSEWORK BREAKDOWN** | | | **TEACHING WEEKLY HOURS** | **ECTS Credits** |
| | | Theory (Lectures) | 2 | |
| | | Practicing Exercises | 1 | |
| | | **TOTAL** | **3** | **3** |
| **COURSE UNIT TYPE** | General Infrastructure | | | |
| **PREREQUISITES** | None | | | |
| **LANGUAGE OF INSTRUCTION/EXAMS** | Greek | | | |
| **COURSE DELIVERED TO ERASMUS STUDENTS** | No | | | |
| **WEB PAGE (URL)** | https://eclass.hmu.gr/courses/ECE102 | | | |

## (2) LEARNING OUTCOMES

**Learning Outcomes**

Upon successful completion of the course, students will be able to use Python programming language to solve scientific problems. In particular, students will be able to:

- know and understand the basic principles of algorithms and programming languages, especially those of dynamic and procedural programming
- understand, interpret and analyze simple scientific problems focusing in particular to the ones encountered in electrical engineering
- Know and understand the basics of both Python and the application development environments available for this language
- design, encode, debug, and run programs in Python to solve such problems

**General Skills**

The graduates of this course will have the following general skills:

- Search, analysis and synthesis of data and information
- Work in an interdisciplinary environment
- Decision making
- Autonomous work

## (3) COURSE CONTENT

- Introductory Concepts (Algorithms, techniques and programming methods, programming languages, compilers and interpreters)
- Introduction to Python (History, philosophy, description, installation and use of language interpreters, examples of simple programs)
- Basic language elements (Values and data types, variables, expressions, operators, comments)
- Numbers and Numerical Functions (Basic operations, integers, floating point numbers, complex numbers)
- Data Input / Output
- Execution Flow Control (Sequential execution, flow control items, if control structures, for loops for, while)
- Functions (Function definition and calling, variable range, documentation strings (docstrings), pure functions and modifier functions, default arguments, anonymous functions, decorators
- Data structures (basic structures, alphanumerics, lists, tuples, dictionaries, sets)
- Files (Access, basic read / write functions, write objects to files)
- Classes and objects (basic concepts, heredity, membership functions
- Exceptions (Mechanism, types, creation and arguments of exceptions)
- Debugging (Types of errors - syntax / logic - use of Python debugger

## (4) TEACHING METHODS - ASSESSMENT

<table>
<tr><td>MODE OF DELIVERY</td><td>In-Class Face-to-Face</td></tr>
<tr><td>USE OF INFORMATION AND COMMUNICATION TECHNOLOGY</td><td>• Use of ICTs in lecturing<br>• Use of ICTs in laboratory-based training<br>• Use of ICTs for the communication with students via the e-class platform<br>• Support of the educational process via the e-class platform</td></tr>
<tr><td>TEACHING ORGANISATION</td><td><table><tr><th>Method description / Activity</th><th>Semester Workload</th></tr><tr><td>Lectures / Tutoring</td><td>39</td></tr><tr><td>Exercises / Project</td><td>26</td></tr><tr><td>Individual Study</td><td>25</td></tr><tr><td>Total Contact Hours</td><td>90</td></tr></table></td></tr>
<tr><td>ASSESSMENT METHODS</td><td>• Exercises / Project 40%<br>• Written Final Exam 60%<br><br>Assessment Language: Greek</td></tr>
</table>

## (5) RECOMMENDED BIBLIOGRAPHY


- *Class notes*
- *Python Εισαγωγή στους υπολογιστές, Α. Αβούρης, Μ. Κουκιάς, Β. Παλιούρας, Κ. Σγάρμπας (ΠΕΚ, 2016).*
- *Εισαγωγή στον αντικειμενοστραφή προγραμματισμό με Python, Κωνσταντίνος Μαγκούτης, Χρήστος Νικολάου (Αποθετήριο "Κάλλιπος", 2016).*
- *Εισαγωγή στον προγραμματισμό με την Python - Νικόλαος Αγγελιδάκης*
- *Python tutorial - Ελληνική Κοινότητα Προγραμματιστών Python*
- *Python Scripting for Computational Science, Hans Petter Langtangen (Heal-Link/Σύνδεσμος Ελληνικών Ακαδημαϊκών Βιβλιοθηκών, 2η έκδοση 2006).*