

COURSE OUTLINE

1. GENERAL

SCHOOL	Engineering		
ACADEMIC UNIT	Mechanical Engineering		
LEVEL OF STUDIES	Undergraduate		
COURSE CODE	FE1200	SEMESTER	1 st
COURSE TITLE	Computer Programming		
INDEPENDENT TEACHING ACTIVITIES <i>if credits are awarded for separate components of the course, e.g. lectures, laboratory exercises, etc. If the credits are awarded for the whole of the course, give the weekly teaching hours and the total credits</i>		WEEKLY TEACHING HOURS	CREDITS
Lectures, Practical Exercises		6	6
<i>Add rows if necessary. The organization of teaching and the teaching methods used are described in detail at (d).</i>			
COURSE TYPE <i>general background, special background, specialized general knowledge, skills development</i>	Core, General Knowledge		
PREREQUISITE COURSES:	None		
LANGUAGE OF INSTRUCTION and EXAMINATIONS:	Greek		
IS THE COURSE OFFERED TO ERASMUS STUDENTS	No		
ΗΛΕΚΤΡΟΝΙΚΗ ΣΕΛΙΔΑ ΜΑΘΗΜΑΤΟΣ (URL)	https://www.mie.uth.gr/?page_id=17684&lang=en		

2. LEARNING OUTCOMES

<p>Learning outcomes <i>The course learning outcomes, specific knowledge, skills and competences of an appropriate level, which the students will acquire with the successful completion of the course are described.</i></p> <p><i>Consult Appendix A</i></p> <ul style="list-style-type: none"> • Description of the level of learning outcomes for each qualifications cycle, according to the Qualifications Framework of the European Higher Education Area • Descriptors for Levels 6, 7 & 8 of the European Qualifications Framework for Lifelong Learning and Appendix B • Guidelines for writing Learning Outcomes <p>The course aims to introduce students to the fundamental principles of computer programming through the Python programming language. Emphasis is placed on structured programming techniques and the methodologies involved in designing and developing algorithms for numerical data processing. By the end of the course, students will be capable of writing efficient and reliable computer programs to solve practical problems in engineering.</p> <p>Upon successful completion of the course, students will be able to:</p> <ul style="list-style-type: none"> • Analyze problem data and design appropriate algorithms for their processing • Implement these algorithms in Python by developing robust and maintainable programs • Create computer programs to solve basic problems in numerical methods • Extend their programming skills to other languages beyond Python with minimal difficulty 								
<p>General Competences <i>Taking into consideration the general competences that the degree-holder must acquire (as these appear in the Diploma Supplement and appear below), at which of the following does the course aim?</i></p> <table style="width: 100%; border: none;"> <tr> <td style="width: 50%; border: none;"><i>Search for, analysis and synthesis of data and information, with the use of the necessary technology</i></td> <td style="width: 50%; border: none;"><i>Project planning and management</i></td> </tr> <tr> <td style="border: none;"><i>Adapting to new situations</i></td> <td style="border: none;"><i>Respect for difference and multiculturalism</i></td> </tr> <tr> <td style="border: none;"><i>Decision-making</i></td> <td style="border: none;"><i>Respect for the natural environment</i></td> </tr> <tr> <td style="border: none;"></td> <td style="border: none;"><i>Showing social, professional and ethical responsibility and</i></td> </tr> </table>	<i>Search for, analysis and synthesis of data and information, with the use of the necessary technology</i>	<i>Project planning and management</i>	<i>Adapting to new situations</i>	<i>Respect for difference and multiculturalism</i>	<i>Decision-making</i>	<i>Respect for the natural environment</i>		<i>Showing social, professional and ethical responsibility and</i>
<i>Search for, analysis and synthesis of data and information, with the use of the necessary technology</i>	<i>Project planning and management</i>							
<i>Adapting to new situations</i>	<i>Respect for difference and multiculturalism</i>							
<i>Decision-making</i>	<i>Respect for the natural environment</i>							
	<i>Showing social, professional and ethical responsibility and</i>							

<i>Working independently</i> <i>Team work</i> <i>Working in an international environment</i> <i>thinking</i> <i>Working in an interdisciplinary environment</i> <i>Production of new research ideas</i>	<i>sensitivity to gender issues</i> <i>Criticism and self-criticism</i> <i>Production of free, creative and inductive</i> <i>.....</i> <i>Others...</i>
<ul style="list-style-type: none"> • Retrieve, analyze and synthesize data and information, with the use of necessary technologies • Autonomous and Team Work • Decision Making • Development of innovative and inductive thinking • Critical Thinking 	

3. SYLLABUS

<p>Fundamental principles of Information Technology</p> <p>Number systems and their representations</p> <p>Algorithm development and design techniques</p> <p>Principles of structured programming</p> <p>Introduction to the Python programming language</p> <p>Fundamental programming constructs:</p> <ul style="list-style-type: none"> • Sequential execution • Conditional (decision) structures • Iterative (repetition) structures <p>Working with NumPy arrays for numerical computing</p> <p>Sorting and searching algorithms</p> <p>Numerical analysis algorithms:</p> <ul style="list-style-type: none"> • Root-finding methods • Numerical integration techniques <p>User-defined data structures and their implementation</p> <p>File handling and data formatting techniques</p> <p>Creating and using user-defined functions</p> <p>Built-in Python functions for processing:</p> <ul style="list-style-type: none"> • Numbers • Strings • Lists • Dictionaries • Files <p>Modules - Packages</p>

4. TEACHING and LEARNING METHODS - EVALUATION

<p>DELIVERY</p> <p><i>Face to face, Distance learning, etc.</i></p>	Face to face	
<p>USE OF INFORMATION AND COMMUNICATIONS TECHNOLOGY</p> <p><i>Use of ICT in teaching, laboratory education, communication with students</i></p>	Use of ICT	
<p>TEACHING METHODS</p> <p><i>The manner and methods of teaching are described in detail.</i></p> <p><i>Lectures, seminars, laboratory practice, fieldwork, study and analysis of bibliography, tutorials, placements, clinical practice, art workshop, interactive teaching, educational visits, project, essay writing, artistic creativity, etc.</i></p>	<p>Activity</p>	<p>Semester workload</p>
	Lectures	70 hours
	Self-evaluating exercises	35 hours
	Autonomous work	45 hours
	Total	150 hours

<p><i>The student's study hours for each learning activity are given as well as the hours of non-directed study according to the principles of the ECTS</i></p>	
<p>STUDENT PERFORMANCE EVALUATION <i>Description of the evaluation procedure Language of evaluation, methods of evaluation, summative or conclusive, multiple choice questionnaires, short-answer questions, open-ended questions, problem solving, written work, essay/report, oral examination, public presentation, laboratory work, clinical examination of patient, art interpretation, other Specifically-defined evaluation criteria are given, and if and where they are accessible to students.</i></p>	<p>I. Written final exams (100%) II. Homework (Bonus: 20%)</p>

5. ATTACHED BIBLIOGRAPHY

- N. Χατζηγιαννάκης, Η γλώσσα Python σε βάθος, 2023, Εκδ. Κλειδάριθμος.
- Αρ. Μπούρας, Γ. Κάππος, Python 3 – Αλγοριθμική και προγραμματισμός, 2020, Εκδ. Κλειδάριθμος.
- Δ. Καρολίδης, Μαθαίνετε εύκολα Python, 2018, Εκδ. Άβακας.
- Er. Matthes, Η γλώσσα προγραμματισμού Python, 2020, Εκδ. Δισίγμα.
- D. Allen, Σκέψου σε Python, 2020, Εκδ. Κλειδάριθμος.