

## MODULE OUTLINE

### 1. GENERAL INFORMATION

<b>SCHOOL</b>	SCHOOL OF SCIENCE AND TECHNOLOGY		
<b>PROGRAM COURSE</b>	INFORMATICS		
<b>LEVEL OF STUDY</b>	UNDERGRADUATE		
<b>MODULE CODE</b>	PLI-10	<b>YEAR OF STUDY</b>	1 <sup>st</sup>
<b>MODULE TITLE</b>	Introduction to Informatics		
<b>INDEPENDENT TEACHING ACTIVITIES</b> <i>in case credits are awarded for separate components/parts of the course, e.g. in lectures, laboratory exercises, etc. If credits are awarded for the entire course, give the weekly teaching hours and the total credits</i>		<b>HOURS</b>	<b>CREDIS</b>
	Weekly teaching hours * 32 weeks	16	18 ECTS
<b>COURSE TYPE</b> <i>Background knowledge, Scientific expertise, General Knowledge, Skills Development</i>	Background knowledge, Scientific expertise Compulsory		
<b>PREREQUISITE MODULES:</b>	During the 1st year of studies if student chooses only one module, then this must be the PLI10 "Introduction to Informatics". If he/she chooses 2 modules it is recommended to choose in addition to PLI10 and PLI12 "Mathematics for Informatics I". If he/she has the required time and wishes to choose three modules he/she can also choose PLI11 "Principles of Software technology".		
<b>LANGUAGE OF INSTRUCTION AND EXAMS</b>	GREEK		
<b>THE MODULE IS OFFERED TO ERASMUS STUDENTS</b>	No (due to annual duration of the module)		
<b>MODULE WEBSITE (URL)</b>	<a href="https://www.eap.gr/education/undergraduate/computer-science/topics/#eisagogi_stin_pli10foriki">https://www.eap.gr/education/undergraduate/computer-science/topics/#eisagogi_stin_pli10foriki</a>  Each module has its own space in the Learning Management System of EAP ( <a href="http://study.eap.gr">http://study.eap.gr</a> ), with controlled access (use of code) for students and teaching staff.		

### 2. LEARNING OUTCOMES

<p><b>Learning Outcomes</b></p> <ul style="list-style-type: none"> <li>• The course learning outcomes, specific knowledge, skills and competences of an appropriate (certain) level, which students will acquire upon successful completion of the course, are described in detail. It is necessary to consult Appendix A:             <ul style="list-style-type: none"> <li>• Description of the Level of Learning Outcomes for each course according to the Qualifications Framework of the European Higher Education Area</li> </ul> </li> </ul>
--

- *Descriptive Indicators of Levels 6, 7 & 8 of the European Qualifications Framework for Lifelong Learning and Annex B.*
- *Summary Guide for writing Learning Outcomes*

## VOLUME 1: INTRODUCTION TO COMPUTER SCIENCE

### a) Knowledge

Upon successful completion of the study of the first volume, students will know:

- the historic evolution of computers and informatics
- the basic subjects of Informatics and its applications
- the computer structure as well as the functionality of its subsystems
- how to store and represent data and information on a computer
- the basic numbering systems used in informatics and their arithmetic operations
- the description of the basic logic gates and circuits
- the concepts of programming, algorithms and programming languages

### b) Understanding

Upon successful completion of the study of the first volume, students will understand:

- the ways of storing, transmitting and processing data
- operations in various arithmetic systems
- the process of representing logical - digital circuits
- algorithmic concepts as well as the concepts of application software and systems

### c) Application

Upon successful completion of the study of the first volume, students will be able to:

- perform value conversions and arithmetic operations on different arithmetic systems
- design logical (digital) sequential circuits

### d) Analysis

Upon successful completion of the study of the first volume, students will be able to:

- describe the various computer subsystems and their functionality
- to categorize the basic subjects of Informatics and its applications

### e) Synthesis

Upon successful completion of the study of the first volume, students will be able to:

- compose logical circuits from gates

### f) Evaluation

Upon successful completion of the study of the first volume, students will be able to:

- evaluate the impact of Informatics in respect to technology evolution and its implications in society

## VOLUME 2: PROGRAMMING TECHNIQUES

### a) Knowledge

Upon successful completion of the study of the second volume, students will know:

- the concept of algorithms, programming practices, program design principles, the criteria of programs' suitability, as well as structured programming principles and programming structures
- advanced programming techniques, such as subroutines, recursion and regression
- software documentation issues and debugging

### b) Understanding

Upon successful completion of the study of the second volume, students will understand:

- how to solve problems, how to design and develop programs
- the concepts of variable, data type, data structure, operator, parameter and expression
- the communication between the main program and its sub-programs

- how sorting and search algorithms work
- the range of variable declaration

### **c) Application**

Upon successful completion of the study of the second volume, students will be able to:

- describe an algorithm through pseudocode
- design algorithms using basic programming practices and program design methodologies
- apply defensive programming

### **d) Analysis**

Upon successful completion of the study of the second volume, students will be able to:

- recognize and categorize the main programming languages

### **e) Synthesis**

- implement sorting and search algorithms
- design algorithms using arrays

### **f) Evaluation**

Upon successful completion of the study of the second volume, students will be able to:

- select the proper programming structures for algorithmic implementations
- evaluate the efficiency of an algorithm

## **VOLUME 3: DATA STRUCTURES**

### **Knowledge**

Upon successful completion of the study of the third volume, students will know:

- The concept of basic data structures, their implementation and its difference from an individual data type
- One-dimensional and two-dimensional arrays (concept, definition, memory representation, import, print, search-sort using various algorithms) have been already described in previous volumes
- The concept of "Dynamic Data Structure", and more specifically:
  - Single Linked Lists, Double Linked Lists, Circular Linked Lists, with several operations such as:
    - (Enter-Delete values, Search-Sort by various algorithms)
- The concepts of Stack and Queue. Static vs Dynamic implementation. Import (enqueue) Export (dequeue)
- The concept of Tree as dynamic data structure
- In more detail: The concepts Binary Tree, Complete Binary Tree, Binary Search Tree, Heap Tree
- Static vs Dynamic representation. Techniques for accessing a binary tree

NOTE:

- The concept of structure (struct) has been described in Volume 4 (Programming languages)
- The concept of simply linked list and its representation in memory has been described in Volume 4 (Programming languages)

### **Understanding**

Upon successful completion of the study of the third volume, students will understand:

- The different ways in which a data structure can be represented in the main memory of the computer
- How important Data Structures are for the development effective algorithms and programs
- The appropriate way of selecting Data Structures in solving algorithmic problems
- Several sorting algorithms and the different ways of data searching using data structures

- Bubble Sort, Merge Sort, Selection Sort algorithms
- Linear Search, Binary Search

### **Application**

Upon successful completion of the study of the third volume, students will be able to:

- Develop the above data structures in C programming language and implement them into C programs
- APPLICATIONS: To solve real-world problems-applications by adopting appropriate Data Structures (Dynamic or Static)
- Develop algorithmic applications with search and sorting techniques using appropriate data structures for each application
- Design variations and/or combinations of different sorting and search algorithms.
- Design modifications, extensions, or combinations of basic algorithms of operations in lists, stacks, queues, binary search trees, and heap tree

### **Analysis**

Upon successful completion of the study of the third volume, students will be able to:

- Categorize data structures and their various applications
- Adopt an analytical way of thinking , meaning that:
  - Will be able to analyze real-world problems and select the appropriate data structures for dealing with it
- Implement programs per use case that require the adoption of the proper data structures (table, list, stack, queue, trees)

### **Synthesis**

Upon successful completion of the study of the third volume, students will be able to:

- Implement programs per use case that require the adoption of the proper data structures (table, list, stack, queue, trees)

### **Valuation**

Upon successful completion of the study of the third volume, students will be able to:

- Design and implement efficient algorithms in parallel with the ability of selecting the appropriate data structures for real-world problem solving
- Be able to experiment with a data structure

## **VOLUME 4: PROGRAMMING LANGUAGES**

### **Knowledge**

Upon successful completion of the study of the fourth volume, students will know::

- The concept and basic principles of Structured Programming
- The basic types of C programming language variables, their use for data representation, as well as their declaration commands
- The need to enter comments into a C program
- The basic libraries of C Programming language
- The declaration of constants in C Programming language
- The structure of a C program, (as a case of a structured programming language)
- The concept “compiler”
- The concepts of logical and syntactic error
- The process of identifying logical and syntactic errors (debugging) using debugging or tracing methods
- The logical operators and logical operations of the C programming language
- Input and output commands scanf, printf and their use
- The implementation of all Algorithmic Selection Structures, in C programming language
- The implementation of all Algorithmic Repetition Structures, in C
- Static data structures. The concept of One/ Two Dimensional Table data structure in C programming language
- The declaration, input and display of a Table’s elements in C

- The string concept
- Searching and Sorting Algorithms in C (Divide and conquer methods)
- Implementation of sorting and searching algorithms on Tables in C
- Working with memory - The concept of Pointer in C Language
- The concept of Function in C
- Developing C programs using functions.
  - The concepts of formal and actual parameters
  - Passing parameters' values (pass by value)
  - Pass by reference using pointers
  - Development of relative examples-cases in C
- Use of recursive functions in C
- The "Struct"
- Using arrays of structures
- Understanding how to use structures and pointers to design and develop a simply linked list and to insert values (Conceptual – Theoretic discussion)
- Design og proper structures (struct) for various cases

#### **Understanding**

Upon successful completion of the study of the fourth volume, students will be able to understand :

- The differences between the three Algorithmic Repetition Structures in C programming language (for, while, do - while) as well as the selection criteria of each in the development of C programs.
- How to declare and develop functions in C programming language
- How to pass variables into functions and return their value in the main program
- Functions of void type in C
- The choice between using Static Data Structures (Tables) or Dynamic Data Structures (Simply Linked Lists)
- Dynamic memory management in C programming language
- The concept of the recursive functions, as well as the advantages / disadvantages over typical functions performing iterations (with loop execution)

#### **Application**

Upon successful completion of the study of the fourth volume, students will be able to:

- Develop solutions to real world or computational problems by developing programs in C
- Describe the operation as well as the results of a simple program in C programming language
- Use a simple programming environment IDE (the Dev-C++ Integrated Development Environment – in Thematic Unit PLH-10)
- Debug programs with logical and syntactic errors using traces or debuggers (especially in the cases of pointers).
- Take advantage of dynamic memory capabilities in C programming language to store dynamically changing data structures
- Pass parameters to functions using pointers

#### **Analysis**

Upon successful completion of the study of the fourth volume, students will be able to:

- Analyze a complex problem in small structural components, each of which will perform a specific function, based on the principles of structured programming
- Design and develop functions of appropriate type, based on the previous analysis
- Select the appropriate algorithmic - programming structures, as well as variables (static or dynamic memory) for the basic functions of their programs

#### **Synthesis**

Upon successful completion of the study of the fourth volume, students will be able to perform:

- Design a program in C programming language that will implement the solution of a real problem, using structured programming techniques

<p><b>Valuation</b></p> <p>Upon successful completion of the study of the fourth volume, students will be able to perform:</p> <ul style="list-style-type: none"> <li>• Provide an estimate of memory and runtime requirements for the functions they implement (either using a recursive or an iterative approach)</li> <li>• Optimize their programming skills and techniques</li> </ul>																
<p><b>General Competences</b></p> <p><i>Taking into consideration the general competences that students/graduates must acquire (as those are described in the Diploma Supplement and are mentioned below), at which of the following does the course attendance aim?</i></p> <table> <tr> <td><i>Search for, analysis and synthesis of data and information by the use of appropriate technologies,</i></td> <td><i>Project planning and management</i></td> </tr> <tr> <td><i>Adapting to new situations</i></td> <td><i>Respect for diversity and multiculturalism</i></td> </tr> <tr> <td><i>Decision-making</i></td> <td><i>Environmental awareness</i></td> </tr> <tr> <td><i>Individual/Independent work</i></td> <td><i>Social, professional and ethical responsibility and sensitivity to gender issues</i></td> </tr> <tr> <td><i>Group/Team work</i></td> <td><i>Critical thinking</i></td> </tr> <tr> <td><i>Working in an international environment</i></td> <td><i>Development of free, creative and inductive thinking</i></td> </tr> <tr> <td><i>Working in an interdisciplinary environment (Other.....citizenship, spiritual freedom, social</i></td> <td><i>.....</i></td> </tr> <tr> <td><i>Introduction of innovative research</i></td> <td><i>awareness, altruism etc.) .....</i></td> </tr> </table>	<i>Search for, analysis and synthesis of data and information by the use of appropriate technologies,</i>	<i>Project planning and management</i>	<i>Adapting to new situations</i>	<i>Respect for diversity and multiculturalism</i>	<i>Decision-making</i>	<i>Environmental awareness</i>	<i>Individual/Independent work</i>	<i>Social, professional and ethical responsibility and sensitivity to gender issues</i>	<i>Group/Team work</i>	<i>Critical thinking</i>	<i>Working in an international environment</i>	<i>Development of free, creative and inductive thinking</i>	<i>Working in an interdisciplinary environment (Other.....citizenship, spiritual freedom, social</i>	<i>.....</i>	<i>Introduction of innovative research</i>	<i>awareness, altruism etc.) .....</i>
<i>Search for, analysis and synthesis of data and information by the use of appropriate technologies,</i>	<i>Project planning and management</i>															
<i>Adapting to new situations</i>	<i>Respect for diversity and multiculturalism</i>															
<i>Decision-making</i>	<i>Environmental awareness</i>															
<i>Individual/Independent work</i>	<i>Social, professional and ethical responsibility and sensitivity to gender issues</i>															
<i>Group/Team work</i>	<i>Critical thinking</i>															
<i>Working in an international environment</i>	<i>Development of free, creative and inductive thinking</i>															
<i>Working in an interdisciplinary environment (Other.....citizenship, spiritual freedom, social</i>	<i>.....</i>															
<i>Introduction of innovative research</i>	<i>awareness, altruism etc.) .....</i>															
<p>Search for, analysis and synthesis of data and information by the use of appropriate technologies,  Adapting to new situations  Decision-making  Individual/Independent work  Project planning and management  Critical thinking  Development of free, creative and inductive thinking</p>																

**3. MODULE CONTENT**

<p>The main objective of the module is to acquaint the student with the basic principles of Computer Science and information management, to gain basic knowledge of problem solving, techniques, tools and structured programming languages, as well as program development principles. Also, the aim is to acquaint knowledge in the use of simple and complex data structures and in the management of information systems, in order to be able to follow the evolutionary course of technology, informatics and their application, now and in the future.</p> <p>The key subjects of the module are:</p> <ol style="list-style-type: none"> <li>Introduction to Computer Science</li> <li>Programming Techniques</li> <li>Data Structures</li> <li>Programming Languages</li> </ol>
--

**4. TEACHING METHODS--ASSESSMENT**

<p style="text-align: center;"><b>MODES OF DELIVERY</b></p> <p><i>Face-to-face, in-class lecturing, distance teaching and distance learning etc.</i></p>	<p>Distance education with five Group Counseling Meetings (OSS) during the academic year on weekends.</p>														
<p style="text-align: center;"><b>USE OF INFORMATION AND COMMUNICATION TECHNOLOGY</b></p> <p><i>Use of ICT in teaching, Laboratory Education, Communication with students</i></p>	<p>We use :</p> <p>Remote meetings tools (skype for business), Presentation software (e.g. power point), Specialized software in the subjects under study (Pseudocode Compiler, Dev-C++).</p> <p>Additionally, the students use office automation tools, web browsers and e-reader for digital books.</p>														
<p style="text-align: center;"><b>MODULE DESIGN</b></p> <p><i>Description of teaching techniques, practices and methods: Lectures, seminars, laboratory practice, fieldwork, study and analysis of bibliography, tutorials, Internship, Art Workshop, Interactive teaching, Educational visits, projects, Essay writing, Artistic creativity, etc</i></p> <p><i>The study hours for each learning activity as well as the hours of selfdirected study are given following the principles of the ECTS.</i></p>	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: center;"><i>Activity</i></th> <th style="text-align: center;"><i>Annual Workload</i></th> </tr> </thead> <tbody> <tr> <td>5 OSS (* 4 hours)</td> <td style="text-align: center;">20</td> </tr> <tr> <td>32 Horizontal tutorial OSS (* 2 hours)</td> <td style="text-align: center;">64</td> </tr> <tr> <td>Preparation of Assignments (4 assignments * 10 hours)</td> <td style="text-align: center;">40</td> </tr> <tr> <td>Examination</td> <td style="text-align: center;">3</td> </tr> <tr> <td>Individual study</td> <td style="text-align: center;">385-449</td> </tr> <tr> <td><b>Total module workload (hours)</b></td> <td style="text-align: center;"><b>512-576</b></td> </tr> </tbody> </table>	<i>Activity</i>	<i>Annual Workload</i>	5 OSS (* 4 hours)	20	32 Horizontal tutorial OSS (* 2 hours)	64	Preparation of Assignments (4 assignments * 10 hours)	40	Examination	3	Individual study	385-449	<b>Total module workload (hours)</b>	<b>512-576</b>
<i>Activity</i>	<i>Annual Workload</i>														
5 OSS (* 4 hours)	20														
32 Horizontal tutorial OSS (* 2 hours)	64														
Preparation of Assignments (4 assignments * 10 hours)	40														
Examination	3														
Individual study	385-449														
<b>Total module workload (hours)</b>	<b>512-576</b>														
<p style="text-align: center;"><b>STUDENT PERFORMANCE EVALUATION/ASSESSMENT METHODS</b></p> <p><i>Detailed description of the evaluation procedures.</i></p> <p><i>Language of evaluation, assessment methods, formative or summative (conclusive), multiple choice tests, short- answer questions, open-ended questions, problem solving, written work, essay/report, oral exam, presentation, laboratory work, other.....etc.</i></p> <p><i>Specifically defined evaluation criteria are stated, as well as if and</i></p>	<p>Elaboration of written assignments during the academic year, the average of the grades of which participates in the formation of the final grade of module by 30%, if there is a passable in the final or repetitive examinations. In the final written exams the grade of the written assignments participates in the formation of the final grade of module by 70%.</p> <p>All the criteria are posted, both in each written assignment (in the LMS study.eap.gr), as well as in the general regulation of HOU at:  <a href="https://www.eap.gr/education/study-regulations/">https://www.eap.gr/education/study-regulations/</a></p>														

where they are accessible by the students	
---	--

## (6) SUGGESTED BIBLIOGRAPHY

*- Suggested bibliography:*

HOU Publications:

Volume A: Introduction to Computer Science, HOU, Patras 2000. PLH10/1/18

Volume B: TProgramming Techniques, HOU, Patras 2008. PLH10/2/18

Volume C: Data Structures, HOU, Patras 2008. PLH10/3/09

Volume D: Programming Languages, HOU, Patras 2000. PLH10/4/18

Additional digital (and multimedia) material is available within the “study” platform.

*-Related scientific Journals:*

- 1) Information Sciences, Elsevier, ISSN: 0020-0255
- 2) Programming and Computer Software, Springer, ISSN: 0361-7688
- 3) Journal of Computational Science, Elsevier, ISSN: 1877-7503
- 4) ACTA Informatica, Springer, ISSN: 0001-5903
- 5) Algorithmica, Springer, ISSN: 0178-4617
- 6) Computing, Springer, ISSN: 0010-485X