

## COURSE DESCRIPTION

### 1. GENERAL

<b>SCHOOL</b>	MUSIC AND AUDIOVISUAL ARTS		
<b>DEPARTMENT</b>	AUDIO AND VISUAL ARTS		
<b>LEVEL</b>	Undergraduate		
<b>COURSE CODE</b>	TEC110	<b>SEMESTER</b>	1 <sup>st</sup>
<b>COURSE TITLE</b>	Introduction to Computer Science I		
<b>INDEPENDENT TEACHING ACTIVITIES</b>	<b>WEEKLY TEACHING HOURS</b>	<b>ECTS</b>	
Lecture, Tutorial	4	6	
<b>COURSE CATEGORY</b>	General Background		
<b>COURSE TYPE</b>	Compulsory		
<b>PREREQUISITES</b>	-		
<b>LANGUAGE OF TEACHING and EXAMINATIONS</b>	Greek		
<b>THE COURSE IS OFFERED TO ERASMUS STUDENTS</b>	YES (In English)		
<b>URL</b>	<a href="https://avarts.ionio.gr/en/studies/undergraduate/courses-descriptions/tec110/">https://avarts.ionio.gr/en/studies/undergraduate/courses-descriptions/tec110/</a>		
<b>ECLASS</b>	<a href="https://opencourses.ionio.gr/courses/DAVA149/">https://opencourses.ionio.gr/courses/DAVA149/</a>		

### 2. TEACHING RESULTS

<b>Teaching Results</b>
<p>The aim of the course is to lay the foundations for understanding the functioning and potential of computer science. The main principles of operation and the structure of computers are briefly covered, while at the same time, the basic chapters concerning the science of Informatics are introduced.</p> <p>The course aims to create the appropriate background for better linking with the subject matter of the following courses in the next semesters related to Information Technology and Programming.</p>
<b>General Skills</b>
<ul style="list-style-type: none"> <li>• Seek, analyze and synthesize data</li> <li>• Autonomous work</li> <li>• Team work</li> <li>• Project design and management</li> <li>• Freedom of thought</li> </ul>

### 3. CONTENT

<p>Introduction to the fundamental concepts of Computer Science. Historical review from the Antikythera Computer to the Quantum Computer. Social, ethical and legal issues related to Information Technology. The Turing and Von Neumann models. Numerical systems. Storing and displaying numbers. Numerical and Logical operations. Description of the internal architecture of the computer. Integrated Circuits. Central Processing Unit, main memory, input/output systems. Computer Networks. Software. Operating Systems. Introduction to the concept of programming. Basic algorithmic structures. Introduction to programming languages: machine language, symbolic language, high-level languages. Presentation of the ARDUINO micro-controller. The course is supported with tutorials.</p> <p>For each week of teaching, the student will be able to ...</p> <p><b>Week 1:</b> Historical &amp; Technological Evolution of Computers (Chapter 1)</p> <ul style="list-style-type: none"> <li>• Provide a history of computers and their technology from Antikytheron analog computers and lamps to transistors and quantum technology.</li> <li>• Define a Turing model of a computer.</li> <li>• Define von Neumann's computer model.</li> </ul>
--

- Describe the three components of a computer (hardware, data, and logic).
- Describe some social and ethical issues related to the use of computers.

## **Week 2:** Numerical Systems (Chapter 2)

- Understand the concept of numerical systems.
- To distinguish the differences between the positional and non-positional numerical systems.
- Describe the binary, decimal, octal and hexadecimal systems.
- Convert a binary, octal, or hexadecimal number to the decimal system.
- Convert a decimal number to the binary, octal, and hexadecimal systems.
- Convert a binary number to octal and vice versa.
- Convert a binary number to hexadecimal and vice versa.
- Calculate the number of digits required in each system to represent a particular value.

## **Week 3:** Data Representation (Chapter 3)

- Identify the five types of data (numbers, text, audio, images, videos) that are used within a computer.
- Describe how different data is stored with corresponding encoding systems (IEEE, ASCII, etc.) inside the computer.

## **Week 4:** Operations with Data (Chapter 4)

- Identify the three categories of operations performed on data.
- Perform single-person and binary logic operations in bit patterns.
- To distinguish logical offset operations from numerical displacements.
- Perform insertion and subtraction operations on integers that are stored as a complement to two.
- Perform operations of addition and subtraction in integers stored in sign and size.
- Perform add-on and subtraction operations in real numbers stored in floating-point format.

## **Week 5:** Computer Organization (Chapter 5)

- Describe the three subsystems of a computer.
- Describe the role of the Central Processing Unit (CPU).
- Describe the phases of recall, decode, and cycle execution.
- Describe the main memory and its address space.
- Define the input / output subsystem.
- Understand how subsystems are interconnected.
- Describe various input / output addressing methods.
- Distinguish the two basic trends in computer design.
- Understand how to improve the performance of a computer by using funneling and parallel processing.

## **Week 6:** Computer Networks (Chapter 6)

- Describe the criteria to be met by networks, physical structures and network categories.
- Describe the TCP / IP protocol collection as an Internet networking model.
- Set TCP / IP protocols collection levels and their relationship.
- Describe the client-server architecture of the Internet.
- Describe three of the first Internet applications: email, file transfer, and remote connection.
- Understand the Web as the most common application of the Internet, as well as its components.
- To distinguish the three types of Internet documents: static, dynamic and active.

## **Week 7:** Operating Systems (Chapter 7)

- Understand the role of the operating system.
- Understand the boot process to load the operating system into memory.
- List the components of an operating system.
- Describe the role of the memory manager, processes, devices, and files in an operating system.
- Understand the basic features of the three most common operating systems (UNIX, Linux and Windows).

## **Week 8:** Algorithms (Chapter 8)

- Define an algorithm and relate it to the solution of a problem.
- Define three structures and describe their use in algorithms.

- Describe the UML charts and the pseudocode as well as how to use them in algorithms.
- Describe basic algorithms and their applications.
- Describe the concept of classification and understand them mechanisms of the three elementary classification algorithms.
- Describe the concept of searching and understand them mechanisms of the two common search algorithms.
- Define sub-algorithms and their relationships with algorithms.
- To distinguish iterative and retrospective algorithms.

**Week 9:** Programming Languages (Chapter 9)

- Describe the evolution of programming languages from machine language to high-level languages.
- Understand how a program in a high-level language translates into machine language.
- Understand the differences between four computer language versions.
- Understand the process model and how it interacts with a program unit and data elements.
- Understand the object-oriented model and how it interacts with a program unit and objects.
- Define the functional model and understand its applications.
- Define the manifest model and understand its applications.
- Understand common concepts of procedural and object-oriented languages.

**Week 10:** ARDUINO (hardware)

- Understand the Arduino system in terms of hardware.

**Week 11:** ARDUINO (software)

- Understand the ARDUINO development platform.

**Week 12:** ARDUINO (applications)

- Understand ARDUINO applications.

**Week 13:** ARDUINO and Interactive Art

- Understand the use of Arduino in Interactive Art facilities.

#### 4. TEACHING AND LEARNING METHODS - EVALUATION

<b>TEACHING METHOD</b>	Lectures												
<b>USE OF INFORMATION AND COMMUNICATION TECHNOLOGIES</b>	Enhanced by multimedia content. The learning process is supported by the asynchronous e-learning platform e-class.												
<b>TEACHING STRUCTURE</b>	<table> <tr> <td>Activity</td> <td>Semester Workload</td> </tr> <tr> <td>Lectures</td> <td>26</td> </tr> <tr> <td>Tutoring Lectures</td> <td>26</td> </tr> <tr> <td>Literature Study and Analysis</td> <td>64</td> </tr> <tr> <td>Practice and Preparation</td> <td>34</td> </tr> <tr> <td><b>Course Total (ECTS: 6)</b></td> <td><b>150</b></td> </tr> </table>	Activity	Semester Workload	Lectures	26	Tutoring Lectures	26	Literature Study and Analysis	64	Practice and Preparation	34	<b>Course Total (ECTS: 6)</b>	<b>150</b>
Activity	Semester Workload												
Lectures	26												
Tutoring Lectures	26												
Literature Study and Analysis	64												
Practice and Preparation	34												
<b>Course Total (ECTS: 6)</b>	<b>150</b>												
<b>EVALUATION OF STUDENTS</b>	Evaluation is done by written examination at the end of the semester.												

#### 5. BIBLIOGRAPHY

1. Introduction to Computer Science, 2nd Edition, Behrouz A. Forouzan, Firouz Mosharraf, 2008, ISBN 960-209-707-8, (550 pages). KLEIDARITHMOS, 2010.
2. THE "COMMAND" OF GALILEO, AN ANTHOLOGY FROM SIGNIFICANT SCIENCE OF THE SCIENTIFIC REASON, Bolles Edmund Blair, University Publication of Crete, 2005.