

COURSE DESCRIPTION

1. GENERAL

SCHOOL	MUSIC AND AUDIOVISUAL ARTS		
DEPARTMENT	AUDIO AND VISUAL ARTS		
LEVEL	Undergraduate		
COURSE CODE	TEC210	SEMESTER	2 nd
COURSE TITLE	Introduction to Computer Science II		
INDEPENDENT TEACHING ACTIVITIES	WEEKLY TEACHING HOURS	ECTS	
Lecture, Tutorial, Hands-on Lab	4	7	
COURSE CATEGORY	General Background		
COURSE TYPE	Compulsory		
PREREQUISITES	-		
LANGUAGE OF TEACHING and EXAMINATIONS	Greek		
THE COURSE IS OFFERED TO ERASMUS STUDENTS	YES (In English)		
URL	https://avarts.ionio.gr/en/studies/undergraduate/courses-descriptions/tec210/		
ECLASS	https://opencourses.ionio.gr/courses/DAVA184/		

2. TEACHING RESULTS

Teaching Results
<p>The course is the continuation of the TEC110 course.</p> <p>The aim of the course is to set the introductory bases for understanding the capabilities of Computer Science. The basic principles of operation and the structure of computers are briefly covered, while at the same time introducing the basic chapters concerning Computer Science, such as Computer Logic, Data Organization and various other advanced topics, e.g. Compression of Data, Computer Security and Computer Theory. The emphasis is on Students learning how to search for new sources of knowledge, apply this knowledge to other subjects of the Department Curriculum, to use it in problem analysis and in the creation or evaluation of Audiovisual Art (interactive, web-based, video, game, etc.).</p>
General Skills
<ul style="list-style-type: none"> • Seek, analyze and synthesize data • Autonomous work • Team work • Project design and management • Freedom of thought

3. CONTENT

<p>Software technology. Software life cycle, development process models, modularity. Data structures. The meaning of table. Records, linked lists. Abstract data types. Trees - binary trees. Graphs. Files structures. Databases and database models. Relations, entities, fields. Base operations. Structured Query Language (SQL). Data compression (image, sound, video). Data security, public and private encryption key. Authentication, key management, digital signature. Calculation theory. Introduction to artificial intelligence (knowledge representation, expert systems, visual processing, pattern recognition). The course is combined with tutorials and labworks on such topics as networks, algorithm development, databases, audio and visual processing, programming in assembly language and high level language.</p> <p>Upon completion of the 13 lectures, the student will be able to ...</p> <p>Week 1: Logic Technology (Chapter 10)</p> <ul style="list-style-type: none"> • Understand the life cycle of software (phases) and development process models. • Identify the quality of the documentation (user & system).
--

- Apply the program build process (writing and debugging, compiling, execution).

Labwork 1: Programming (using Matlab)

Week 2: Data Structures (Chapter 11)

- Define a data structure.
- Define arrays as data structures and describe how they are used to store data item lists.
- To distinguish the name of an array from the names of its elements.
- Describe the functions defined for arrays.
- Define records as data structures and describe how they are used to store properties belonging to simple data items.
- To distinguish the name of a record from the names of its fields.
- Define a linked list as a data structure and describe how it is implemented using indexes.
- Understand the mechanism through which the nodes of an array are accessed.
- Describe the functions defined for the linked lists.
- Compare and compare arrays, records, and linked lists.
- Define array, record and linked list applications.

Labwork 2: INTEL-8086 simulator presentation.

Week 3: Abstract Data Types (Chapter 12)

- Understand the concept of abstract data type.
- Define the stacks, the basic functions that are performed on them, their applications, and how they are implemented.
- Define the queues, the basic functions performed on them, their applications, and how they are implemented.
- Define the general line lists, the basic functions that are performed on them, their applications, and how they are implemented.
- Define general trees and their applications.
- Define binary trees - a specific tree type - and their applications.
- Define binary search trees (BST) and their applications.
- Define the graphs and their applications.

Labwork 3: Assembly program exercises on the INTEL-8086 simulator.

Week 4: File Structures (Chapter 13)

- Define the two categories of file access methods: sequential access and random access.
- Understand the structure of sequential files and the way they are updated.
- Understand the structure of indexed files and the relationship between the index and the data file.
- Understand the concept of fragmented files and describe some methods of fragmentation.
- Describe address conflicts and how to solve them.
- Understand the concept of directories and how they can be used to organize files.
- To distinguish text files and binary files.

Labwork 4: HTML - Presentation of the basic document format of the web, example implementation.

Week 5: Databases (Chapter 14)

- Define a database and database management system (DBMS) as well as describe the components of a DBMS.
- Describe the architecture of a DBMS based on the ANSI / SPARC standard.
- Define the three traditional database models: hierarchical, network, and relational.
- Describe the relational model and relationships.
- Understand the functions that apply to a relational SQL-based database.
- Describe the steps taken in database design.
- Define ERM and E-R charts and explain the entities and relationships in this model.
- Define the hierarchical levels of normalization and understand the logic used for normalizing relationships.
- Describe other types of databases other than the relational model.

Labwork 5: Databases - Presentation of environment.

Week 6: Data Compression (Chapter 15)

- Understand the difference between non-loss and loss of compression.
- Describe current length coding and how compression is achieved with this method.
- Describe Huffman coding and how compression is achieved with this method.
- Describe the Lempel Ziv encoding and the dictionary's role in coding and decoding.
- Describe the basic concept behind the JPEG standard for image compression.
- Describe the basic concept behind the MPEG standard for video compression and its relation to the JPEG standard.
- Describe the basic concept behind the MP3 standard for audio compression.

Labwork 6: Relational Base Example and creation in OpenOffice (SQL queries).

Week 7: Security Encryption (Chapter 16)

- Define the three security objectives - confidentiality, integrity, and availability - as well as types of attacks against these goals.
- Define the five security services - data confidentiality, data integrity, authentication, non-denial, and access control - that prevent security attacks.
- Describe two techniques for providing security services: cryptography and segregation.
- Understand the differences between symmetric key cryptography and asymmetric key cryptography and describe how confidentiality is ensured using cryptographic symmetric key or asymmetric key algorithms.
- Describe how integrity is ensured by using cryptographic segmentation functions.
- Understand the idea of digital signatures and how they can ensure the integrity and authentication of messages and non-denial.
- Describe briefly the entity's authentication certification and the categories of testimonies: something that is known, something that is possessed, and something inherent.
- Describe the four techniques used for entity authentication: the password-based technique, the challenge-response technique, the zero-knowledge technique, and the biometrics.
- Describe key management.

Labwork 7: Programming - Familiarization in the language environment.

Week 8: Calculation Theory (Chapter 17)

- Describe the programming language we call Simple Language and define its basic commands.
- Writing macros in the Simple Language with a combination of simple commands.
- Describe the components of a Turing machine as a computational model.
- To show how to simulate Simple Language simple commands using a Turing machine.
- Understand Church-Turing and its importance.
- Define the Gödel number and its implementation.
- Understand the concept of the termination problem and how it turns out that this problem is unsolvable.
- To distinguish solvable from unsolvable problems.
- To distinguish the polynomials from the non-polynomial solvable problems.

Labwork 8: Programming - examples.

Week 9: Artificial Intelligence (Chapter 18).

- Define Artificial Intelligence and give it a brief history.
- Describe how to represent knowledge in an intelligent agent.
- Show how expert systems can be used when a specialist person is unavailable.
- Explain how an artificial agent can be used to simulate work done by humans.
- Explain how experienced systems and shared systems can use different search techniques to solve problems.
- Explain how a learning process in humans can be simulated to some extent by using neural networks that create electronically equivalent neurons called artificial neurons.

Labwork 9: Programming examples.

Week 10: Repeat ARDUINO (use emulator).

Labwork 10: ARDUINO - presentation of the platform, examples.

Week 11: Progress Review.

Labwork 11: ARDUINO - Material Exercises.

Week 12: Progress Solutions - Lesson Evaluation Questionnaire.

Labwork 12: ARDUINO - Software exercises.

Week 13: Interactive Art Applications.

Labwork 13: ARDUINO & Interactive Art.

4. TEACHING AND LEARNING METHODS - EVALUATION

TEACHING METHOD	Lectures																
USE OF INFORMATION AND COMMUNICATION TECHNOLOGIES	Enhanced by multimedia content. The learning process is supported by the asynchronous e-learning platform e-class.																
TEACHING STRUCTURE	<table> <thead> <tr> <th></th> <th>Semester Workload</th> </tr> </thead> <tbody> <tr> <td>Activity</td> <td></td> </tr> <tr> <td>Lectures</td> <td>13</td> </tr> <tr> <td>Lab Practice</td> <td>26</td> </tr> <tr> <td>Tutoring Lectures</td> <td>13</td> </tr> <tr> <td>Literature Study and Analysis</td> <td>80</td> </tr> <tr> <td>Practice and Preparation</td> <td>43</td> </tr> <tr> <td>Course Total (ECTS: 7)</td> <td>175</td> </tr> </tbody> </table>		Semester Workload	Activity		Lectures	13	Lab Practice	26	Tutoring Lectures	13	Literature Study and Analysis	80	Practice and Preparation	43	Course Total (ECTS: 7)	175
	Semester Workload																
Activity																	
Lectures	13																
Lab Practice	26																
Tutoring Lectures	13																
Literature Study and Analysis	80																
Practice and Preparation	43																
Course Total (ECTS: 7)	175																
EVALUATION OF STUDENTS	<p>The following methods may be used:</p> <p>Progress Test. Bp = degree of progress.</p> <p>Theory and exercises: written examination (with open books). Be = Degree of examination.</p> <p>Laboratory exercises: laboratory test per person. Total grade = 70% grade of written examination ($B\pi * 0,1 + B\varepsilon * B\varepsilon * 0,9$) + 30% grade of laboratory</p> <p>Note 1: The total grade is entered in the student record by the Secretariat if both the grade of the written examination and the laboratory grade are \geq of the base (i.e., 3.5 & 1.5 respectively).</p> <p>Note 2: The software used are MATLAB, Microsoft Access and ARDUINO. Also assembly language, assembler and the microcontroller ARDUINO are employed.</p>																

5. BIBLIOGRAPHY

1. How Computers Work, Ron White, Key lock. ISBN: 978-960-461-194-2 (304 pages)

2. THE "COMMAND" OF GALILEO, AN ANTHOLOGY FROM SIGNIFICANT SCIENCE OF THE SCIENTIFIC REASON, Bolles Edmund Blair, University Publication of Crete, 2005.