COURSE OUTLINE

(1) GENERAL

SCHOOL	ENGINEERING		
ACADEMIC UNIT	ELECTRICAL AND COMPUTER ENGINEERING DEPT.		
LEVEL OF STUDIES	Undergraduate		
COURSE CODE	**ECE_K650**	**SEMESTER**	**6**
COURSE TITLE	**Introduction to Distributed Systems**		

INDEPENDENT TEACHING ACTIVITIES *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*	**WEEKLY TEACHING HOURS**	**CREDITS**
Lectures	2	
Seminars / Practice exercises	1	
Laboratory	1	
Add rows if necessary. The organisation of teaching and the teaching methods used are described in detail at (4).	4	5

COURSE TYPE *general background, special background, specialised, general knowledge, skills development*	Special background
PREREQUISITE COURSES:	There are no prerequisite courses. Recommended background knowledge: Operating Systems, Functional Programming, Object-Oriented Design and Programming
LANGUAGE OF INSTRUCTION and EXAMINATIONS:	Greek
IS THE COURSE OFFERED TO ERASMUS STUDENTS	Yes
COURSE WEBSITE (URL)	https://www.ece.uop.gr/

(2) LEARNING OUTCOMES

Learning outcomes

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.

Consult Appendix A

- *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*

The course aims to present the basic concepts of distributed processing and the principles and main problems of distributed systems (DS) so that students are able to understand the corresponding technologies and services and the important advantages they offer.

Keywords: distributed memory, distributed computation model, clock synchronization, wave and traversal algorithms, distributed routing, leader election, mutual exclusion, P2P networks

Learning Outcomes

Upon successful completion of the course, student will be able to:

At the knowledge level:

- Understand the meaning of the distributed system
- Know the types and techniques of communication in DS.
- Understand the concepts of distributed computing and its application in highly critical scientific areas of informatics.
- Understand the problem of clock synchronization and know the most important synchronization techniques.
- Know types and techniques of information dissemination in DS
- To know the types and the most important techniques of distributed routing.
- Understand the problem of election
- Understand the problem of distributed mutual exclusion
- To know the most important properties, categories and characteristics of peer-to-peer (P2P) systems
- Be aware of distributed application development environments / libraries and modern programming languages that offer the distributed feature.

At the skill level:

- Understand the modern DS.
- Solve the problem of clock synchronization in a distributed environment
- Solve the problem of information dissemination in a distributed environment.
- Select the most appropriate distributed routing algorithms.
- To be able to handle security and fault issues in DS.
- Design and implement efficient, correct and robust distributed protocols
- Explain the potential of peer-to-peer systems and the advantages of using them.

At the level of abilities:

- Use efficiently integrated environments for distributed applications development.
- Be able to provide efficient and secure solutions to complex business problems using distributed systems.
- Be able to understand and apply the distributed principles of control and processing to a multitude of issues of increased interest such as large-scale data, machine learning, embedded systems, the internet of things, etc.

General Competences

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?

Search for, analysis and synthesis of data and information, with the use of the necessary technology	*Project planning and management*
Adapting to new situations	*Respect for difference and multiculturalism*
Decision-making	*Respect for the natural environment*
Working independently	*Showing social, professional and ethical responsibility and sensitivity to gender issues*
Team work	*Criticism and self-criticism*
Working in an international environment	*Production of free, creative and inductive thinking*
Working in an interdisciplinary environment	*……*
Production of new research ideas	*Others…*
	…….

- Search for, analysis and synthesis of data and information, with the use of the necessary technology
- Working independently
- Team Work
- Working in an international environment
- Production of free, creative and inductive thinking
- Production of new research ideas

(3) SYLLABUS

The course is developed in the following 13 lectures:

1. Basic Concepts of Distributed Systems, Distributed Processing, Middlewear Software, Middlewear Software Services. Distributed Systems Organization
2. Communication in Distributed Systems, Remote Procedure Calls, Remote Objects, Messaging Oriented Communication, Stream Oriented Communication.
3. The distributed computing model. DS topologies. Criteria and performance evaluation metrics.
4. Introduction/use of integrated environment for distributed applications development. Communication, definition of topology, design, implementation and execution of distributed code.
5. The problem of clock synchronization. Clock Synchronization algorithms. Logical time and time stamps. Global state and distributed snapshots.
6. Dissemination of information to DS. Wave and traversal algorithms.
7. Use of integrated environment for distributed applications development. Design and implementation of distributed applications that refer to clock synchronization, wave and traversal algorithms.
8. The problem of distributed routing. Kategories and routing algorithms for DS.
9. The leader election problem. Symmetry breaking in DS. Probabilistic DS algorithms. The problem of mutual exclusion in DS. Centralized and decentralized techniques.
10. Use of integrated environment for distributed applications development. Design and implementation of distributed applications that refer to distributed routing, leader selection and mutual exclusion.
11. P2P systems. Properties, applications and categories. Distributed data management in peer-to-peer networks.
12. Fault tolerance, robust DS, error recovery, self-stabilized systems, security issues in DS.
13. Modern categories of distributed systems: Introduction to large-scale data management systems, sensor systems, mobile networks, Cloud Computing and Internet of Things.

(4) TEACHING and LEARNING METHODS - EVALUATION

DELIVERY *Face-to-face, Distance learning, etc.*	Face to face in class and in the laboratory. Distance learning support via e-Class system.
USE OF INFORMATION AND COMMUNICATIONS TECHNOLOGY *Use of ICT in teaching, laboratory education, communication with students*	• Supporting the learning process through the e-Class platform (for notification of the course regulations, for distribution of slides, laboratory exercises, supplementary material, announcements, links, bibliography, etc.). • During the lectures of the theoretical part, a projector and presentations in electronic form are used, which are also posted on the eclass from the beginning of the semester. • During the lectures, a computer is used to write and execute code. • Use of specialized software integrated development environment (such asMPI, MQTT, Netbeans, etc.).

TEACHING METHODS

The manner and methods of teaching are described in detail.

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.

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

Activity	***Semester workload***
Theory Lectures	26
Seminars / Practice exercises	13
Laboratory Exercises	13
Preparation of laboratory exercises	13
Projects preparation	25
Independent study of lectures and bibliography	35
Course Total	***125 hours (5 ECTS)***

STUDENT PERFORMANCE EVALUATION

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.

A. Written final exam that includes:

- Solve exercises
- Multiple choice questions
- Short answer questions

B. Preparation of laboratory exercises and projects.

Remarks:

- The final grade results from the weighting of the theory and work grades with coefficients determined at the beginning of the semester and announced to the students via e-class. Indicatively it will be about 60% - 40%
- Laboratory exercises and assignments will be submitted electronically and students will be asked to take an oral exam on them.
- The exam material and the evaluation process are communicated to the students in the lecture hall and in the e-class.

(5) ATTACHED BIBLIOGRAPHY

- Suggested bibliography:

- Tanenbaum A.S., Van Steen M., Κατανεμημένα Συστήματα: Αρχές και Παραδείγματα, 1η έκδοση, Εκδόσεις Κλειδάριθμος, 2006.
- Κάβουρας Ι.Κ., Μήλης Ι.Ζ., Ξυλωμένος Γ.Β., Ρουκουνάκη Α.Α., Κατανεμημένα Συστήματα με Java, 3η έκδοση, Εκδόσεις Κλειδάριθμος, 2011.
- Coulouris G., Dollimore J., Kindberg T., Blair G., Distributed Systems, Concepts and Design, 5th ed.,Addison-Wesley, 2011.
- Introduction to Distributed Algorithms, G. TEL, Cambridge University Press.
- N. Lynch, Distributed Algorithms, Morgan Kaufman.

- Related academic journals:

- ACM Journal of Parallel and Distributed Computing
- ACM International Journal of Parallel, Emergent and Distributed Systems
- IEEE Transactions on Parallel and Distributed Systems