Course unit title:	**Computer Architecture**		
Course unit code:	AEEE294		
Type of course unit:	Compulsory		
Level of course unit:	Bachelor (1st Cycle)		
Year of study:	2		
Semester when the unit is delivered:	3		
Number of ECTS credits allocated :	5		
Name of lecturer(s):	Dr. Haris Haralambous		
Learning outcomes of the course unit:	1. Describe the metrics and benchmarks based on which evaluations between different systems can be made. 2. Describe the instruction execution cycle with reference to the flow of information at the register level, and analyse typical Instruction Set Architectures with respect to the number of operands, addressing modes and branch types. 3. Describe the internal structure and operation of a CPU datapath and design a simple single-cycle and a multi-cycle non-pipelined CPU. 4. Explain how memory organisation affects the performance of a computer and how cache memory exploits locality to reduce the memory wall problem. 5. Describe the operation and evaluate the performance of the common cache memory mapping methods. 6. Explain the operation of a pipelined CPU and the advantages gained from such an approach. Analyse the problems that arise with respect to hazards and practical ways to detect and overcome these hazards.		
Mode of delivery:	Face-to-face		
Prerequisites:	AEEE192	Co-requisites:	None
Recommended optional program components:	None		
Course contents:	• **CPU Performance:** Overview of the history of computer architecture development. Emerging trends and technology drivers. Assessing computer performance based on different metrics (execution time, CPI and other performance parameters).Amdahl’s law. • **Introduction to computer organization and architecture:** Instruction cycle and flow of information at the register level. Instruction Set Architectures, instruction formats and instruction decoding. Relation between machine language, assembly language and high level languages. • **CPU design basics:** Datapaths, register files, ALU, shift and rotate circuits. Control unit implementation, hardwired control. Single-cycle and multi-cycle non-pipelined CPU design. • **Computer Arithmetic:** Implementation of a basic 32-bit ALU. Control signals for the ALU. Multiplication and Division algorithms. Introduction to floating-point numbers. IEEE double precision floating point format. • **Memory Hierarchy:** The memory locality principle. Cache memory organization and mapping. Cache replacement and write policies. Cache performance metrics. Virtual memory.		

Recommended and/or required reading:	
Textbooks:	Paterson, Hennessy, *Computer Organization and Design: the Hardware/Software Interface*, Morgan Kaufman, 2008
References:	M. Mano, C. R. Kime, *Logic and Computer Design Fundamentals*, Prentice Hall, 2010
Planned learning activities and teaching methods:	The taught part of course is delivered to the students by means of lectures, conducted with the help of computer presentations. Lecture notes and presentations are available through the web for students to use in combination with the textbooks.
Assessment methods and criteria:	• Tests: 40% • Final Exam 60%
Language of instruction:	English
Work placement(s):	No