COURSE PROGRAMME

1. Information about the programme

1.1 University	**University "Alexandru Ioan Cuza" of Iasi**
1.2 Faculty	**Faculty of Computer Science**
1.3 Department	**Department of Computer Science**
1.4 Domain	**Computer Science**
1.5 Cycle	**Licence**
1.6 Programme / Qualification	**Computer Science (in English)**

2. Information about the course

2.1 Course Name			Computer Architecture and Operating Systems				
2.2 Course taught by			Lecturer PhD. VLAD TUDOR RADULESCU				
2.3 Seminary / laboratory taught by			Lecturer PhD. VLAD TUDOR RADULESCU				
2.4 Year	I	**2.5** Semester	I	**2.6** Type of evaluation*	E	**2.7** Course type**	Ob

**E – Exam / C – Colloquium / V – Verification*

***OB – Obligatory / OP – Optionally / F – Facultative*

3. Total hours (estimated per semester and activities)

3.1 Number of hours per week	4	**3.2** course	2	**3.3** seminary/laboratory	2
3.4 Total number of hours	56	**3.5** course	28	**3.6** seminary/laboratory	28

Distribution	hours
Individual study using textbooks, course notes, bibliography items, etc.	30
Supplimentary study (library, on-line platforms, etc.)	15
Individual study for seminary/laboratory, homeworks, projects, etc.	15
Tutoring	5
Examination	4
Other activities	0

3.7 Total hours of individual activity*	69
3.8 Total hours per semester	125
3.9 Credit points	5

4. Pre-requisites - Curriculum (if necessary)

-

5. Conditions (if necessary)

5.1 Course	-
5.2 Seminary / Laboratory	-

6. Objectives

Understanding the underlying principles and hardware/software technologies of the computing systems.

7. Specific competencies/Learning outcomes

- The student/graduate identifies, explains and uses fundamental concepts on data structures, algorithms and programming paradigms, as well as computer architecture.
- The student/graduate develops, implements and proves complex software solutions using efficient algorithms and various programming paradigms.
- The student/graduate coordinates tehnical teams for the development of computer applications, by taking responsible decision about their optimization and integration.

8. Contents

8.1 Course	**Teaching methods**	**Remarks** (number oh hours, references)
Introduction. Boole functions	exposition, debate, case studies, exercises	-
Minimization. Combinational circuits	exposition, debate, case studies, exercises	-
Sequential circuits	exposition, debate, case studies, exercises	-
Internal representations. Fixed-point representations	exposition, debate, case studies, exercises	-
Fixed-point representations. Overflows. Floating-point representations	exposition, debate, case studies, exercises	-
Floating-point representations. Computer architecture and organization	exposition, debate, case studies, exercises	-
The memory. Memory hierarchy. Cache memory	exposition, debate, case studies, exercises	-
Recapitulation	exposition, debate, case studies, exercises	-
The Central Processing Unit. Pipeline	exposition, debate, case studies, exercises	-
Improving the CPU performance	exposition, debate, case studies, exercises	-
Parallel architectures. Peripheral devices	exposition, debate, case studies, exercises	-
The interrupt system. The operating system. Process management	exposition, debate, case studies, exercises	-

8.1 Course	Teaching methods	Remarks (number oh hours, references)
Memory management	exposition, debate, case studies, exercises	-
Virtual memory. Creation and execution of programs	exposition, debate, case studies, exercises	-

Bibliography
J. L. Hennessy, D. A. Patterson, Computer Architecture - A Quantitative Approach, Morgan Kaufmann Publishers, 1990. D. A. Patterson, J. L. Hennessy, Computer Organization & Design: The Hardware/Software Interface, Morgan Kaufmann Publishers, 1998. R. W. Hockney, C. R. Jesshope, Parallel Computers 2, IOP Publishing, 1988. A. Tanenbaum, Structured Computer Organization, Prentice Hall, 1999. A. Tanenbaum, Modern Operating Systems, Prentice Hall, 2001.

8.2 Seminary / Laboratory	Teaching methods	Remarks (number oh hours, references)
Introduction. Conversions between number bases	exposition, debate, exercises	-
Boole functions. Minimization	exposition, debate, exercises	-
Combinational circuits	exposition, debate, exercises	-
Latches and flip-flops	exposition, debate, exercises	-
Fixed-point representations	exposition, debate, exercises	-
Floating-point representations	exposition, debate, exercises	-
Introduction to the Intel x86 assembly language	exposition, debate, exercises	-
Recapitulation	exposition, debate, exercises	-
Arithmetic and logical instructions	exposition, debate, exercises	-
Jump instructions	exposition, debate, exercises	-
Stack-handling instructions. Function calls	exposition, debate, exercises	-
Arrays and pointers. Structures	exposition, debate, exercises	-
Floating-point instructions	exposition, debate, exercises	-
Recapitulation	exposition, debate, exercises	-

Bibliography
J. L. Hennessy, D. A. Patterson, Computer Architecture - A Quantitative Approach, Morgan Kaufmann Publishers, 1990. Intel® 64 and IA-32 Architectures Software Developer Manuals: http://www.intel.com/content/www/us/en/processors/architectures-software-developer-manuals.html

9. Coordination of the contents with the expectations of the community representatives, professional associations and relevant employers in the corresponding domain

This discipline presents the basic elements of a computing architecture. Such knowledge is essential for understanding the way computers work, regardless of the purpose of their use; particularly, writing efficient programs is not possible without being acquainted with the mechanisms and techniques that are taught here.

10. Assessment and examination

10.1 Continuous assessment	Percentage (min. 30%)	80

Course	Assessment type			
	Percentage		0	
	Failure to pass the continuous assessment results in failure to pass the final assessment			
	Assessment methods	Details	Percentage	with reexamination
Seminary / Laboratory	Assessment type		Practical assessment	
	Percentage		100	
	Failure to pass the continuous assessment results in failure to pass the final assessment		No	
	Assessment methods	Details	Percentage	with reexamination
		Test	50	Yes
		Practical Test	50	No

10.2 Final assessment	Percentage (max. 70%)	20
	Assessment type	Final written assessment

10.3 Special notes (special situations is assessment)	
-	

10.4 Minimum performance standard
Basic knowledge of the concepts related to the hardware structure of the computing systems. The ability to develop low-complexity programs in the assembly language of the Intel x86 processor family.

Date,

Course coordinator,
Lecturer PhD. VLAD TUDOR RADULESCU

Seminary coordinator,
Lecturer PhD. VLAD TUDOR RADULESCU

Aproval date in the department,

Head of the departament,
Assoc. Prof. PhD. ANDREI ARUSOAIE