MINISTRY OF NATIONAL EDUCATION

TECHNICAL UNIVERSITY
OF CLUJ-NAPOCA

SYLLABUS

1. Study Program

1.1	Higher Education Institute	Technical University of Cluj-Napoca
1.2	Faculty	Electronics, Telecommunications and Information Technology
1.3	Department	Communications
1.4	Study domain	Electronics and Telecommunications Engineering
1.5	Study level	Master
1.6	Study program/ Qualification	Telecommunications/ Master
1.7	Type of education	IF (Full-time learning)
1.8	Discipline code	TC11.20

2. Discipline

2.1	Discipline name						Advanced Computer Architectures				
2.2	Subject area						Electronics and Telecommunications Engineering				
2.3	Responsible						Assistant Professor Ovidiu Buza, Ph.D. Ovidiu.Buza@com.utcluj.ro				
2.4	Titular						Assistant Professor Ovidiu Buza, Ph.D.				
2.5	Year of study	I	2.6	Semester	2	2.7	Evaluation	Exam	2.8	Type of discipline	

3. Total estimated time

Year/ Sem	Discipline name	No. of weeks	Course [hours/week]				Applications [hours/week]				Indiv. study	TOTAL	ECTS
			C	S	L	P		S	L	P			
I/2	Advanced Computer Architectures	14	2	0	1	0	28	0	14	0	58	100	4

3.1	Number of hours per week	3	3.2	course	2	3.3	applications	1
3.4	Total hours per curriculum	42	3.5	course	28	3.6	applications	14
Individual study								Hours
Study based on manuals, course materials, references and notes								14
Supplementary documentation in libraries, electronic platforms and on field								10
Preparation of seminars/laboratories, homeworks, essays, portfolios								10
Tutorial work								8
Assesments								4
Other activities								12

3.7	Total hours of individual study	58
3.8	Total hours per semester	100
3.9	ECTS	4

4. Prerequisites (if necessary)

4.1	Curriculum	
4.2	Competences	

5. Requisites (if necessary)

5.1	Course	Video-projector, screen, whiteboard
5.2	Applications	Local network with Internet access, Visual C++, PVM, Condor

6. Specific competences acquired

Professional competences	Theoretical knowledge (What do the student should know)	The students will acquire knowledge about: - basic concepts of computer architecture; - methods for computer performance evaluation; - advanced techniques in designing computer central units; - architecture of computer systems: principles, current and perspective developments; - high performance architectures; parallel and distributed architectures; - operating systems and programming standards for parallel architectures
	Acquired skills (What the student is able to do)	The students will be able to: - implement programs in Visual C under Windows 32-bit; - use programming techniques based on events and messages; - use structures and specific classes for programming under Windows 32-bit; - set up and programming a parallel virtual machine; - implement various logical topologies on a parallel virtual machine; - realise of multitask programs on the parallel virtual machine; - work with and implement concurrent and parallel programming techniques; - implement parallel algorithms; - work with grid computing techniques
	Acquired abilities (what equipment/ instruments/ softwares the student is able to handle)	The students will be able to use: - Visual C under Windows 32-bit; - parallel programming environments like PVM and Condor
Transversal competences		CT3 Adapting to new technologies, professional and personal development through continuing education using electronic documentation and printed sources, in Romanian and in at least one international language (English). Competencies for analysis and synthesis and optimization systems thinking. Flexibility in thinking and ability to work with interdisciplinary concepts and tools.

7. Discipline objectives (based on the grid of specific competences acquired)

7.1	General objective	Acquiring knowledge in the field of computer architectures
7.2	Specific objectives	- acquiring basic and specific knowledge of computer architecture; - knowledge of the current and future principles in computer systems design; - knowledge of high-performance, parallel and distributed computer architectures; - acquiring knowledge about operating systems and programming standards for parallel architectures; - knowing how to realise and work with a parallel virtual machine; - acquiring basic knowledge about Grid computing

8. Contents

8.1. Course (titles)		Teaching methods	Observations
1	Introduction. History; virtual machine; languages; taxonomy of computer architectures; processor families; performance evaluation methods	Presentation, discussions	Videoprojector
2	Computer basic architecture. The central unit, memory, buses, controllers and components, input/ output devices		
3	Advanced techniques for central unit architecture design. The pipelined superscalar architecture; NetBurst architecture; Pentium processors family		
4	High performance architectures. Vector processors; MIMD and SIMD processors; RISC architectures; SPARC architectures		
5	Memory Systems. Types of memory; memory performance indicators; multiple memory units; associative memory; cache; virtual memory; memory modules design		
6	Interconnection networks. Direct networks; indirect networks; circuit-switching; packet switching; information routing techniques		
7	Parallel and distributed architectures. Multiprocessor architecture; transputers; hypercubes; distributed systems; grid architecture; OCCAM specification		
8	SIMD systems. Array processors, vectorial processors, systolic systems		
9	Multicomputers. Organization; message passing; massively parallel systems; COW multicomputers		
10	Multiprocessor systems. Structure, consistency models for shared memory, network connections for multiprocessors		
11	Multiprocessors with uniform memory access (UMA). Specifications; symmetric multiprocessors; UMA multiprocessors based on grid switches		
12	Multiprocessors with non-uniform memory access (NUMA). Specifications; NC_NUMA multiprocessors; CC_NUMA multiprocessors; COMA multiprocessors		
13	Operating systems and programming techniques for multiprocessors; concurency exploitation, detection of parallelism inside programs, synchronization mechanisms, examples		
14	Standards and programming environments for parallel architectures. MPI standard, PVM environment, OCCAM language, intelligent agents		
8.2. Applications (laboratory work)		Teaching methods	Observations
1	The defining elements of 32-bit programming	Programming, experiments	PC network
2	Structures and classes used in Windows 32-bit programming		
3	Introduction to PVM; building a parallel virtual machine		
4	Functions for message passing and task control		
5	Functions for processes groups in PVM		
6	Implementation of Cannon's algorithm using PVM library (I)		
7	Implementation of Cannon's algorithm using PVM library (II)		
8	Introduction to grid computing		
9	Programs execution in Condor (I)		
10	Programs execution in Condor (II)		
11	Workflows in Condor		
12	Other examples of parallel algorithms		
13	Applications on parallel architectures		
14	Laboratory test		

References:

1. J. L. Hennessy, D. A. Patterson , *Computer Architecture, Fifth Edition: A Quantitative Approach* (The Morgan Kaufmann Series in Computer Architecture and Design), Elsevier, 2012, ISBN-10: 012383872X

2. G. Lerman, L. Rudolph, *Parallel Evolution of Parallel Processors* (Evaluation in Education and Human Services), Springer, 2013, ISBN-13: 978-1461362371
3. D. B. Kirk, W. W. Hwu, *Programming Massively Parallel Processors, Second Edition: A Hands-on Approach*, Elsevier, 2012, ISBN-10: 0124159923
4. Shane Cook , *CUDA Programming: A Developer's Guide to Parallel Computing with GPUs* (Applications of GPU Computing Series), Elsevier, 2013, ISBN-10: 0124159338
5. C. Lin, L. Snyder, *Principles of Parallel Programming*, Addison-Wesley, 2008, ISBN-13: 978-0321487902
6. G. Sebestyen, *Informatică industrială*, Ed. Albastră, Cluj-Napoca, 2006
7. Z.F.Baruch, *Structura sistemelor de calcul*. Editura Albastră, Cluj-Napoca, 2005
8. B.B.Brey, Intel 32-Bit Microprocessor: 80386, 80486 & Pentium, Prentice Hall; 7thEd 2005
9. D.Gorgan, G. Sebestyen, *Proiectarea calculatoarelor*, Ed. Albastră, Cluj-Napoca, 2005
10. D.A.Patterson, J.L.Hennessy, *Computer Organization and Design: The Hardware/ Software Interface,* 3rd Edition , Morgan Kaufmann Publishers, August 2004
11. D.E. Comer, *Essentials of Computer Architecture*, Prentice Hall; US edition, August 2004

9. Discipline content corroborated with the expectations of the epistemic community representatives, associations, professional and related program employers

Acquired skills will be needed in the following possible COR occupations: electronics engineer, telecommunications engineer, system and computer design engineer, or new occupations proposed to be included in COR (network operating engineer, test engineer, traffic engineer, communications system consultant).

10. Assessment

Type of activity	10.1	Evaluation criteria	10.2	Evaluation method	10.3	The weight of the final grade
Course		Exam (E = 1...10)		Written test		50%
		Scientific essay (S = 1…10)		Scientific essay presented by each student		25%
Applications		Laboratory test (L = 1 ... 10)		Written test at the end of semester		25%
10.4 Minimum performance standard						
The final grade (N) is calculated as follows: N = 0,5E+0,25L+0,25S The condition for obtaining the ECTS credits is that all components of the final grade to be higher than or equal to 5 (five).						

Date
24.06.2018

Titular
Assistant Professor
Ovidiu Buza, Ph.D.

Responsible
Assistant Professor
Ovidiu Buza, Ph.D.

Date of approval
24.06.2018

Head of Department
Professor Virgil Dobrota, Ph.D.