SYLLABUS

1. Data about the program of study

1.1 Institution	The Technical University of Cluj-Napoca
1.2 Faculty	Faculty of Automation and Computer Science
1.3 Department	Computer Science
1.4 Field of study	Computer Science and Information Technology
1.5 Cycle of study	Bachelor of Science
1.6 Program of study/Qualification	Computer science/ Engineer
1.7 Form of education	Full time
1.8 Subject code	37.

2. Data about the subject

2.1 Subject name	***Structure of Computer Systems***				
2.2 Course responsible/lecturer	Prof. dr. eng. Gheorghe Sebestyen – Gheorghe.Sebestyen@cs.utcluj.ro				
2.3 Teachers in charge of seminars/ laboratory/ project	S.l.dr.eng. Anca Hangan, As.dr.eng. Madalin Neagu, drd. Vlad Miclea				
2.4 Year of study	III	2.5 Semester	2	2.6 Type of assessment (E - exam, C - colloquium, V - verification)	E
2.7 Subject category	*DF – fundamentală, DD – în domeniu, DS – de specialitate, DC – complementară*				DD
	DI – Impusă, DOp – opțională, DFac – facultativă				DI

3. Estimated total time

3.1 Number of hours per week	5	of which:	Course	2	Seminars		Laboratory	2	Project	1
3.2 Number of hours per semester	70	of which:	Course	28	Seminars		Laboratory	28	Project	14

3.3 Individual study:	
(a) Manual, lecture material and notes, bibliography	14
(b) Supplementary study in the library, online and in the field	4
(c) Preparation for seminars/laboratory works, homework, reports, portfolios, essays	13
(d) Tutoring	0
(e) Exams and tests	3
(f) Other activities:	0

3.4 Total hours of individual study (suma (3.3(a)...3.3(f)))	34
3.5 Total hours per semester (3.2+3.4)	104
3.6 Number of credit points	4

4. Pre-requisites (where appropriate)

4.1 Curriculum	Digital system design, Computer architecture
4.2 Competence	Understand and operate with basic concepts regarding computer system's hardware

5. Requirements (where appropriate)

5.1. For the course	
5.2. For the applications	

6. Specific competence

6.1 Professional competences	**C2** – Designing hardware, software and communication components (5 credits) **C2.1** – Describing the structure and functioning of computational, communication and software components and systems **C2.2** – Explaining the role, interaction and functioning of hardware, software and communication components **C2.3** – Building the hardware and software components of some computing systems using algorithms, design methods, protocols, languages, data structures, and technologies

	C2.4 – Evaluating the functional and non-functional characteristics of the computing systems using specific metrics **C2.5** – Implementing hardware, software and communication systems
6.2 Cross competences	N/A

7. Discipline objective (as results from the *key competences gained*)

7.1 General objective	The main goal of the course is to present in an accessible way advanced design methods and techniques used in today's microprocessors and computer systems
7.2 Specific objectives	To study: Methods and metrics for computer performance assessment Advanced CPU designs (pipelining, multicore, parallele and distributed computing) Memory hierarchies: cache memory, virtual memory, new DRAM technologies RISC architecture Parallel computers architectures – hardware issues and solutions

8. Contents

8.1 Lectures	Hours	Teaching methods	Notes
Introduction. Computer Performance Parameters and Methods of Improvement	2	Lecture based on slides	
Computer performance and optimality, Benchmarking	2		
The Arithmetical and Logical Unit (ALU)	2		
The Central Processing Unit (CPU) – MIPS architecture, pipeline, hazard cases	2		
The Central Processing Unit – advance techniques: Scoreboard method, Tomasulo's algorithm, Branch prediction techniques	2		
The Central Processing Unit – multi-core systems	2		
Microprocessors – basic components and advanced implementations	2		
Memory System – memory technologies (SRAM, DRAM) and design principles	2		
Memory Hierarchies – cache and virtual memory	2		
Interconnection Systems – serial and parallel synchronous and asynchronous buses, multipoint interconnections	2		
Parallel Computer Architectures - different levels of parallel execution	2		
RISC Architectures – principles and implementation examples	2		
Distributed Computing – GRID and Cloud Systems	2		
Technological Perspectives in Computer Architectures	2		

Bibliography

1. Gorgan Dorian, Sebestyen Gheorghe, Structura Calculatoarelor, Editura albastra, Cluj-Napoca 2005
2. Hennessy John, Patterson David, Computer architecture, a Quantitative Approach, Ed. Elsevier, 2007
3. Baruch, Z. F., Structure of Computer Systems, U.T.PRES, Cluj-Napoca, 2002, ISBN 973-8335-44-2.

8.2 Applications – Seminars/Laboratory/Project	Hours	Teaching methods	Notes
Measuring the performance of computer systems with benchmarks	2	Practical designs, experiments and results assesment	
CPU performance monitoring using the Time-Stamp Counter register	2		
Programming elements in VHDL	2		
Design of ALU components	2		
FPGA Synthesis	2		
Introduction to using PicoBlaze microcontroller with the Nexys3 board	2		
Implementation of a MIPS processor in VHDL - 1	2		
Implementation of a MIPS processor in VHDL - 2	2		
Implementation of a pipelined MIPS processor in VHDL	2		
Memory design - 1	2		
Memory design - 2	2		

Advanced Hardware Design Techniques	2		
Design implementations on NEXYS 3 board	2		
Laboratory Colloquy	2		
Topics for Project Assignments: Implementation of arithmetic circuits; Design and implementation of processors and controllers; Signal Processing; Hardware implementation of DSP and image processing algorithms; Design of I/O interfaces.			
Bibliography Laboratory works at http://users.utcluj.ro/~ancapop/scs.html			

[*] Se vor preciza, după caz: tematica seminariilor, lucrările de laborator, tematica și etapele proiectului.

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

10. Evaluation

Activity type	Assessment criteria	Assessment methods	Weight in the final grade
Course	Theoretical knowledge level	Written exam	60%
Seminar			
Laboratory	Hardware Design skills	Practical evaluation	40%
Project			
Minimum standard of performance: Minimum 5 for the Course and for the Application assessment Grade calculus: 30% midterm + 20% laboratory + 20% project + 30% final exam Conditions for participating in the final exam: Laboratory ≥ 5, Project ≥ 5 Conditions for promotion: final exam ≥ 5			

Course responsible
Prof. dr. eng. Gheorghe Sebestyen

Head of department
Prof.dr.eng. Rodica Potolea

SYLLABUS

1. Data about the program of study

1.1 Institution	The Technical University of Cluj-Napoca
1.2 Faculty	Faculty of Automation and Computer Science
1.3 Department	Computer Science
1.4 Field of study	Computer Science and Information Technology
1.5 Cycle of study	Bachelor of Science
1.6 Program of study/Qualification	Computer science/ Engineer
1.7 Form of education	Full time
1.8 Subject code	38.

2. Data about the subject

2.1 Subject name	***Formal Languages and Translators***				
2.2 Course responsible/lecturer	Assoc.prof. dr.eng. Emil Şt. Chifu – emil.chifu@cs.utcluj.ro				
2.3 Teachers in charge of seminars/ laboratory/ project	Ing. Mihai Anton Cerghizan				
2.4 Year of study	III	2.5 Semester	2	2.6 Type of assessment (E - exam, C - colloquium, V - verification)	E
2.7 Subject category	*DF – fundamentală, DD – în domeniu, DS – de specialitate, DC – complementară*				DD
	DI – Impusă, DOp – opțională, DFac – facultativă				DI

3. Estimated total time

3.1 Number of hours per week	4	of which:	Course	2	Seminars		Laboratory	2	Project	
3.2 Number of hours per semester	56	of which:	Course	28	Seminars		Laboratory	28	Project	
3.3 Individual study:										
(a) Manual, lecture material and notes, bibliography										17
(b) Supplementary study in the library, online and in the field										7
(c) Preparation for seminars/laboratory works, homework, reports, portfolios, essays										16
(d) Tutoring										5
(e) Exams and tests										3
(f) Other activities:										0

3.4 Total hours of individual study (suma (3.3(a)…3.3(f)))	48
3.5 Total hours per semester (3.2+3.4)	104
3.6 Number of credit points	4

4. Pre-requisites (where appropriate)

4.1 Curriculum	Computer Programming, Data Structures and Algorithms
4.3 Competence	Basic knowledge of programming and data structures (preferably in the C language)

5. Requirements (where appropriate)

5.1. For the course	N/A
5.2. For the applications	Computers, specific software

6. Specific competence

6.1 Professional competences	**C1 –** Operating with basic Mathematical, Engineering and Computer Science concepts (2 credits) **C1.1 –** Recognizing and describing concepts that are specific to the fields of calculability, complexity, programming paradigms, and modeling computational and communication systems **C1.2 –** Using specific theories and tools (algorithms, schemes, models, protocols, etc.) for explaining the structure and the functioning of hardware, software and communication systems

	C1.3 – Building models for various components of computing systems **C1.4 –** Formal evaluation of the functional and non-functional characteristics of computing systems **C1.5 –** Providing a theoretical background for the characteristics of the designed systems **C3** – Problems solving using specific Computer Science and Computer Engineering tools (2 credits) **C3.1 –** Identifying classes of problems and solving methods that are specific to computing systems **C3.2 –** Using interdisciplinary knowledge, solution patterns and tools, making experiments and interpreting their results **C3.3 –** Applying solution patterns using specific engineering tools and mehods **C3.4 –** Comparatively and experimentaly evaluation of the alternative solutions for performance optimization **C3.5 –** Developing and implementing informatic solutions for concrete problems
6.2 Cross competences	N/A

7. Discipline objective (as results from the *key competences gained*)

7.1 General objective	- To know the phases, components, and algorithms used by typical language translators. - To provide a formal basis for the development of concepts relating to lexical and syntactic processors in translators.
7.2 Specific objectives	- To know the underlying formal models such as finite state automata and push-down automata, and to understand their connection to language definition through regular expressions and grammars. - To understand the relationships between formal descriptions of the automata in the formal language theory and their practical implementations as lexical and syntactic analyzers in translators. - To know the classes of languages for which a deterministic parser can be implemented. - To describe the syntax of languages to be implemented by using grammars and regular expressions. - To design, develop and test a software project, by utilizing specialized software tools (parser generators), in order to arrive at a translator for an artificial language. - To master and control the phenomena of ambiguity and nondeterminism (conflicts) which occur when using parser generators and lexical analyzer generators.

8. Contents

8.1 Lectures	Hours	Teaching methods	Notes
Descriptive tools: strings and rewriting systems, grammars.	2	- The main ideas with multimedia tehniques - Details and examples at the blackboard, in interaction with the students - There are consultation hours - Students are invited to collaborate in research projects	
Descriptive tools: derivations and parse trees.	2		
Regular grammars and finite automata: finite automata.	2		
Regular grammars and finite automata: state diagrams and regular expressions.	2		
Context-free grammars and pushdown automata: pushdown automata.	2		
Top-down analysis and LL(*k*) grammars: LL(*k*) grammars	2		
Top-down analysis and LL(*k*) grammars: the LL(*k*) algorithm	2		
Top-down analysis and LL(*k*) grammars: elimination of left recursion, left factoring.	2		
LL parsers: strong LL(*k*) grammars, the LL(1) parsing algorithm.	2		

LL parsers: the LL(1) parsing algorithm, computation of FIRST and FOLLOW sets.	2		
Bottom-up analysis and LR(*k*) grammars: situations and closure of a nonterminal, the LR(*k*) algorithm.	2		
Bottom-up analysis and LR(*k*) grammars:the LR(*k*) algorithm.	2		
LR parsers: the LR(0) parsing algorithm.	2		
LR parsers: LR(0) states.	2		

Bibliography

1. W.M. Waite and G. Goos, Compiler Construction, Springer-Verlag, 1984.
2. I.A. Leţia and E.Şt. Chifu, Limbaje formale şi translatoare, Ed. Casa cărţii de ştiinţă, 1998.
3. A.V. Aho, R. Sethi, and J.D. Ullman, Compilers: Principles, Techniques and Tools, Addison-Wesley, 1986.

8.2 Applications – Seminars/Laboratory/Project	Hours	Teaching methods	Notes
Lexical analyzer for C.	2	Brief presentation at the blackboard, implementing and testing homeworks on the computer, individual assignment on the computer.	
The generator of lexical analyzers Lex: Lex source, Lex regular expressions, Lex actions, ambiguous rules, Lex source definitions.	2		
Lex generator: left context sensitivity, examples.	2		
The bottom-up parser generator Yacc: basic specifications, Yacc syntax, actions, lexical analysis, how the parser works.	2		
Yacc generator: ambiguity and conflicts, precedence and associativity, error handling, the Yacc environment, hints for preparing specifications.	2		
Yacc generator: support for arbitrary value types, examples (expression evaluator).	2		
Yacc/ Lex applications: interpreter for a language operating on lists.	2		
Yacc/ Lex applications: interpreter for a language operating on binary trees.	2		
Yacc/ Lex applications::interpreter for a language operating on matrices.	2		
Yacc/ Lex applications: code generator for an imperative language.	2		
Yacc/ Lex test	2		
Building recursive-descent (RD) parsers: expression parser.	2		
RD parsers: parser for a language operating on binary trees.	2		
RD parsers: parser for a language operating on lists.	2		

Bibliography

1. The Lex & Yacc Page, http://www.combo.org/lex_yacc_page/
2. I.A. Leţia, D. Marcu, B. Ungureanu, Procesoare de limbaje. Îndrumător de laborator, Universitatea Tehnică din Cluj-Napoca, 1995.

**Se vor preciza, după caz: tematica seminariilor, lucrările de laborator, tematica și etapele proiectului.*

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

It is a specialty course in Computer Science, its syllabus being both classical and modern. It teaches the students with the basic principles in the design of interpreters and translators for artificial languages. The syllabus of the course has been discussed with other important universities and companies from Romania, Europe, and USA. This syllabus has been evaluated by Romanian governmental agencies (CNEAA and ARACIS).

10. Evaluation

Activity type	Assessment criteria	Assessment methods	Weight in the final grade
Course	- Problem-solving skills - Attendance, Activity	- Written exam	55%
Seminar			

Laboratory	- Problem-solving skills - Attendance, Activity	- Assessment of the Yacc/ Lex activity and test - Assessment of the RD activity and written exam	30% 15%
Project			
Minimum standard of performance: Modeling a typical engineering problems using the domain specific formal apparatus. Grade calculus: 45% lab + 55% final exam Conditions for participating in the final exam: lab ≥ 5 Conditions for promotion: grade ≥ 5			

Course responsible
Assoc.prof. dr. eng. Emil Chifu

Head of department
Prof.dr.eng. Rodica Potolea

SYLLABUS

1. Data about the program of study

1.1 Institution	The Technical University of Cluj-Napoca
1.2 Faculty	Faculty of Automation and Computer Science
1.3 Department	Computer Science
1.4 Field of study	Computer Science and Information Technology
1.5 Cycle of study	Bachelor of Science
1.6 Program of study/Qualification	Computer science/ Engineer
1.7 Form of education	Full time
1.8 Subject code	39.

2. Data about the subject

2.1 Subject name	***Management and communication***				
2.2 Course responsible/lecturer	Assist. Prof. Veronica Maier, PhD				
2.3 Teachers in charge of seminars/ laboratory/ project	-				
2.4 Year of study	III	2.5 Semester	2	2.6 Type of assessment (E - exam, C - colloquium, V - verification)	C
2.7 Subject category	*DF – fundamentală, DD – în domeniu, DS – de specialitate, DC – complementară*				DD
	DI – Impusă, DOp – opțională, DFac – facultativă				DI

3. Estimated total time

3.1 Number of hours per week	2	of which:	Course	2	Seminars		Laboratory		Project	
3.2 Number of hours per semester	28	of which:	Course	28	Seminars		Laboratory		Project	

3.3 Individual study:	
(a) Manual, lecture material and notes, bibliography	6
(b) Supplementary study in the library, online and in the field	7
(c) Preparation for seminars/laboratory works, homework, reports, portfolios, essays	6
(d) Tutoring	3
(e) Exams and tests	2
(f) Other activities:	

3.4 Total hours of individual study (suma (3.3(a)…3.3(f)))	24
3.5 Total hours per semester (3.2+3.4)	52
3.6 Number of credit points	2

4. Pre-requisites (where appropriate)

4.1 Curriculum	Not the case
4.4 Competence	Not the case

5. Requirements (where appropriate)

5.1. For the course	The existence of multimedia equipment
5.2. For the applications	Not the case

6. Specific competence

6.1 Professional competences	**C5 –** Designing, managing the lifetime cycle, integrating and ensuring the integrity of hardware, software and communication systems (1 credit) **C5.1 -** Specifying the relevant criteria regarding the lifetime cycle, quality, security and computing system's interaction with the environment and human operator **C5.2 -** Using interdisciplinary knowledge for adapting an information system to application domain requirements **C5.3 -** Using fundamental principles and methods for security, reliability and

	usability assurance of computing systems **C5.4 -** Adequate utilization of quality, safety and security standards in information processing **C5.5 -** Realization of a project including problem identification and analysis, design and development, while proving the understanding of the basic quality needs and requirements
6.2 Cross competences	**CT2 –** Identifying, describing and conducting processes in the projects management field, undertaking different team roles and clearly and concisely describing own profesional results, verbally or in writing, in Romanian and in an international language. (1 credit)

7. Discipline objective (as results from the *key competences gained*)

7.1 General objective	Understand, assimilate and use of basic management and communication concepts, principles and techniques
7.2 Specific objectives	Understand the basic managerial functions, the organization's internal environment, the motivation of people, the organizational communication, communication barriers, increasing communication, overcoming of internal conflicts and the link between leadership and communication.

8. Contents

8.1 Lectures	Hours	Teaching methods	Notes
Introduction in management: management concept, managerial functions, the managers, challenges confronting contemporary management	2h	multimedia presentation, interactivity by exemplifying the presented concepts, using the questions-answer method during the course, discussing case studies, playing thematic strategy game, interactive lectures	
Organization's internal environment: culture and business ethics	2h		
External environment (general and task environment components)	4h		
Planning	2h		
Organizing	2h		
Coordinating and motivating people	2h		
Controlling and performance assessment	2h		
Organizational communication (content, functions, types, networks); interpersonal and group communications	2h		
Communication barriers	2h		
Increasing the effectiveness of communication	2h		
Conflict and conflict management	2h		
Negotiation and assertive communication	2h		
Leadership and communication	2h		

Bibliography

1. Catana D., Dobra Constantinescu A., Management in Power Point, UTPRES 2004
2. Adler, R.B., Elmhorst,J.M, Communicating at work, Principles and practices for Business and the professions, 8th ed., McGraw Hill, 2005
3. Becker, E.F., Wortmann, J., Mastering communication at work: how to lead, manage and influence, McGraw Hill Professional, 2009, disponibil pe: http://books.google.ro/books/about/Mastering_Communication_at_Work.html?id=0G6LuTp6XhsC&redir_esc=y
4. Bell, A.H., Smith, D.M., Management communication, 2nd ed., John Wiley&Sons Inc., 2006
5. Nicolescu, O. Fundamentele managementului organizaţiei, Editura Universitară, 2008
6. Nicolescu, O., Verboncu, I. Managementul organizaţiei, Editura Economică, Bucureşti, 2007

8.2 Applications – Seminars/Laboratory/Project	Hours	Teaching methods	Notes
-			

Bibliography

-

[*] *Se vor preciza, după caz: tematica seminariilor, lucrările de laborator, tematica și etapele proiectului.*

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

In order to provide the content for the lecture and establish the method of the teaching / learning process the professor organizes meetings with entrepreneurs, who speak to students about the needs and expectations of employers in the area.

10. Evaluation

Activity type	Assessment criteria	Assessment methods	Weight in the final grade
Course	The students answer to open ended and closed questions; involvement during the course by preparing and presenting teamwork papers.	Written exam	100%
Seminar			
Laboratory			
Project			
Minimum standard of performance: Requirement for the credits: N>5			

Course responsible
Assist. Prof. Veronica Maier, PhD

Head of department
Prof.dr.eng. Rodica Potolea

SYLLABUS

1. Data about the program of study

1.1 Institution	The Technical University of Cluj-Napoca
1.2 Faculty	Faculty of Automation and Computer Science
1.3 Department	Computer Science
1.4 Field of study	Computer Science and Information Technology
1.5 Cycle of study	Bachelor of Science
1.6 Program of study/Qualification	Computer science/ Engineer
1.7 Form of education	Full time
1.8 Subject code	40.

2. Data about the subject

2.1 Subject name	***Image processing***
2.2 Course responsible/lecturer	Prof dr. eng. Sergiu Nedevschi (Sergiu.Nedevschi@cs.utcluj.ro)
2.3 Teachers in charge of seminars/ laboratory/ project	Assoc. Prof. dr. eng. Florin Oniga, Assist. Prof. dr. eng. Ion Giosan, Assist. Prof. dr. eng. Raluca Brehar, {Florin.Oniga, Ion.Giosan, Raluca.Brehar}@cs.utcluj.ro

2.4 Year of study	III	2.5 Semester	2	2.6 Type of assessment (E - exam, C - colloquium, V - verification)	E

2.7 Subject category	*DF – fundamentală, DD – în domeniu, DS – de specialitate, DC – complementară*	DD
	DI – Impusă, DOp – opțională, DFac – facultativă	DI

3. Estimated total time

3.1 Number of hours per week	5	of which:	Course	2	Seminars		Laboratory	2	Project	1
3.2 Number of hours per semester	70	of which:	Course	28	Seminars		Laboratory	28	Project	14

3.3 Individual study:	
(a) Manual, lecture material and notes, bibliography	14
(b) Supplementary study in the library, online and in the field	3
(c) Preparation for seminars/laboratory works, homework, reports, portfolios, essays	14
(d) Tutoring	0
(e) Exams and tests	3
(f) Other activities:	0

3.4 Total hours of individual study (suma (3.3(a)…3.3(f)))	34
3.5 Total hours per semester (3.2+3.4)	104
3.6 Number of credit points	4

4. Pre-requisites (where appropriate)

4.1 Curriculum	N/A
4.5 Competence	Computer programming (C++), Data structures and algorithms, Linear Algebra, Numerical methods, Special mathematics.

5. Requirements (where appropriate)

5.1. For the course	Blackboard, video projector, computer
5.2. For the applications	Workstations, specific software (Visual Studio, Diblook)

6. Specific competence

6.1 Professional competences	**C6 -** Designing intelligent systems **C6.1 -** Describing the components of intelligent systems **C6.2 -** Using domain-specific tools for explaining and understanding the functioning of intelligent systems **C6.3 -** Applying the fundamental methods and principles for specifying solutions for typical problems using intelligent systems **C6.4 -** Choosing criteria and methods for the evaluation of quality, performances and limitations of information systems

	C6.5 - Developing and implementing professional projects for intelligent systems
6.2 Cross competences	N/A

7. Discipline objective (as results from the *key competences gained*)

7.1 General objective	Understanding the concepts related to digital images, computer vision and image processing. Learning and applying image processing methods, and designing specific applications.
7.2 Specific objectives	▪ Learning, evaluation and use of image processing specific concepts, algorithms and methods: digital image formats, camera model, statistical analysis, image filtering, image enhancing and restauration, segmentation, measurement. ▪ Acquiring the capacity of finding optimal solutions for image processing algorithm implementation, taking into consideration time and hardware constraints. ▪ Acquiring the capacity of quantitative and qualitative assessement of results, algorithms and systems for image processing. ▪ Learning the use of programming tools and image processing frameworks (Diblook, MS MFC, OPEN CV)

8. Contents

8.1 Lectures	Hours	Teaching methods	Notes
Computer vision and its applications. Structure and functionality of computer vision systems. Image acquisition systems.	2	Interactive teaching, using oral presentations supported by multimedia tools, consultations, involving students in research and development activities.	
Camera model, the image formation process, coordinate transforms, calibration.	2		
Fundamentals of stereovision, stereo configurations, depth computation, epipolar geometry.	2		
Binary image processing: Simple Geometric Properties	2		
Binary image processing: Labeling, Contour Tracing, Polygonal Approximation	2		
Binary image processing: Mathematical Morphology	2		
Grayscale image processing. Statistical properties. Image quality enhancement.	2		
Grayscale image processing: Convolution and Fourier Transform	2		
Grayscale image processing: Noise in images	2		
Grayscale image processing: Digital filtering.	2		
Grayscale image segmentation: Edge based segmentation	2		
Grayscale image processing: Advanced methods for edge extraction and linking.	2		
Grayscale image processing: Textures.	2		
Image region segmentation using intensity, color and texture features.	2		

Bibliography

1. R. C. Gonzales, R. E. Woods, "Digital Image Processing-Second Edition",3rd Edition, *Prentice Hall, 2008*
2. R. C. Gonzalez, R. E. Woods, S. L. Eddins, “Digital Image Processing Using MATLAB”, 2nd ed., *Gatesmark Publishing*, 2009.
3. E. Trucco, A. Verri, “Introductory Techniques for 3-D Computer Vision”, *Prentice Hall, 1998*.
4. G. X.Ritter, J.N. Wilson, “Handbook of computer vision algorithms în image algebra”, *CRC Press*, 2001.
5. S. Nedevschi, T. Marita, R. Danescu, F. Oniga, R. Brehar, I. Giosan, S. Bota, A. Ciurte, V. Andrei, Image Processing – Laboratory Guide, *UTPRES*, Cluj-Napoca, 2016

Online

1. S. Nedevschi, “Prelucrarea imaginilor - Note de curs”, ftp.utcluj.ro/pub/users/nedevschi/IP_2016/

8.2 Applications – Seminars/Laboratory/Project	Hours	Teaching methods	Notes
Laboratory			
Getting started with the DIBLook framework	2	Presentation using the blackboard and multimedia tools.	
The color model. Color-grayscale and grayscale-black&white conversions	2		

The histogram of intensity levels	2	Experiments and implementation using specific software tools (MS Visual Studio, Diblook) Evaluation of the design and implementation phases.	
Geometrical features of binary objects	2		
Binary objects labeling	2		
Border tracing algorithm.	2		
Morphological operations on binary images	2		
Statistical properties of grayscale images	2		
Image filtering in the spatial and frequency domains	2		
Noise modeling and digital image filtering	2		
Edge detection (1)	2		
Edge detection (2)	2		
Region-based image segmentation	2		
Evaluation	2		
Project			
Choosing and discussing the project subject (weeks 1 and 2).	1		
Discussing the literature study and the work schedule (weeks 3 and 4).	1		
Algorithm design (weeks 5 and 6)	1		
Presentation of algorithm implementation. Intermediary evaluation (weeks 7 and 8).	1		
Algorithm testing and validation. Quantitative and qualitative evaluation (weeks 9 and 10).	1		
Algorithm optimization (weeks 11 and 12).	1		
Final project assessment (weeks 13 and 14).	1		
Bibliography 1. S. Nedevschi, T. Marita, R. Danescu, F. Oniga, R. Brehar, I. Giosan, S. Bota, A. Ciurte, V. Andrei, “Image Processing – Laboratory Guide”, *UTPRES*, Cluj-Napoca, 2016 Online http://users.utcluj.ro/~igiosan/teaching_ip.html			

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

The subject is part of the Computer Science and Information Technology curriculum, its contents combining fundamental and practical aspects used in the field of visual information processing (an ever growing domain). The subject content is correlated with the specific curricula of other Universities, in Romania and abroad, and is evaluated by government agencies (CNEAA and ARACIS). The subject’s activities are meant to make the students familiar with the applications and the research directions of the image processing field, helped by the internationally renowned experience of the teachers.

10. Evaluation

Activity type	Assessment criteria	Assessment methods	Weight in the final grade
Course	Testing the theoretical knowledge acquired, and the practical abilities of problem solving.	Written exam	50%
Seminar			
Laboratory	Testing the practical abilities of designing and implementing solutions to specific problems. Attendance and activity.	Lab assessment, project assessment	50%
Project			
Minimum standard of performance: Modeling and implementation of solutions to specific engineering problems, using the domain’s formal apparatus. Grade calculus: 25% laboratory + 25% project + 50% final exam Conditions for participating in the final exam: Laboratory ≥ 5, Project ≥ 5 Conditions for promotion: final exam ≥ 5			

Course responsible	Head of department
Prof. dr. ing. Sergiu Nedevschi	Prof.dr.eng. Rodica Potolea

SYLLABUS

1. Data about the program of study

1.1 Institution	The Technical University of Cluj-Napoca
1.2 Faculty	Faculty of Automation and Computer Science
1.3 Department	Computer Science
1.4 Field of study	Computer Science and Information Technology
1.5 Cycle of study	Bachelor of Science
1.6 Program of study/Qualification	Computer science/ Engineer
1.7 Form of education	Full time
1.8 Subject code	41.

2. Data about the subject

2.1 Subject name	***Software design***				
2.2 Course responsible/lecturer	Prof.dr.eng. Mihaela Dinsoreanu – mihaela.dinsoreanu@cs.utcluj.ro				
2.3 Teachers in charge of seminars/ laboratory/ project	Prof.dr.eng. Mihaela Dinsoreanu				
2.4 Year of study	III	2.5 Semester	2	2.6 Type of assessment (E - exam, C - colloquium, V - verification)	E
2.7 Subject category	*DF – fundamentală, DD – în domeniu, DS – de specialitate, DC – complementară*				DS
	DI – Impusă, DOp – opțională, DFac – facultativă				DI

3. Estimated total time

3.1 Number of hours per week	5	of which:	Course	2	Seminars		Laboratory	2	Project	1
3.2 Number of hours per semester	70	of which:	Course	28	Seminars		Laboratory	28	Project	14

3.3 Individual study:	
(a) Manual, lecture material and notes, bibliography	10
(b) Supplementary study in the library, online and in the field	5
(c) Preparation for seminars/laboratory works, homework, reports, portfolios, essays	10
(d) Tutoring	4
(e) Exams and tests	5
(f) Other activities:	

3.4 Total hours of individual study (suma (3.3(a)...3.3(f)))	34
3.5 Total hours per semester (3.2+3.4)	104
3.6 Number of credit points	4

4. Pre-requisites (where appropriate)

4.1 Curriculum	Programming Techniques, Software Engineering
4.6 Competence	

5. Requirements (where appropriate)

5.1. For the course	Video projector (compulsory), internet connected computer (optional)
5.2. For the applications	16 internet connected computers

6. Specific competence

6.1 Professional competences	**C3** - Problem solving using specific Computer Science and Computer Engineering tools **C3.1** Identifying classes of problems and solving methods that are specific to computing systems **C3.2** Using interdisciplinary knowledge, solution patterns and tools, making experiments and interpreting their results **C3.3** Applying solution patterns using specific engineering tools and methods **C3.4** Evaluating, comparatively and experimentally, the available alternative solutions for performance optimization

	C3.5 Developing and implementing software solutions for specific problems
6.2 Cross competences	N/A

7. Discipline objective (as results from the *key competences gained*)

7.1 General objective	Understand and model requirements, analyse and design appropriate solutions
7.2 Specific objectives	• Identify the most relevant functional and non-functional requirements of a software system and to document them • Design and motivate software architecture for (large scale) software systems • Recognize and apply major software architectural styles, design patterns, and frameworks • Describe a software architecture using various documentation approaches and architectural description languages • Generate architectural alternatives for a problem and select among them

8. Contents

8.1 Lectures	Hours	Teaching methods	Notes
Introduction and basic concepts review	2	Face-to-Face lecture, Powerpoint slides	
Architectural Styles (Structural)	2		
Architectural Styles (Distributed)	2		
Business logic architectural patterns	2		
Data Access and hybrid architectural patterns	2		
Presentation and Concurrency architectural patterns	2		
Midterm exam	2		
Applying Creational Design Patterns	2		
Applying Structural Design Patterns	2		
Applying Behavioral Design Patterns	2		
Class Design Principles (SOLID, GRASP)	2		
Package design Principles	2		
Service oriented architectures	2		
Software Design Quality metrics and final review	2		

Bibliography

1. Ian Gorton, Essential Software Architecture, Springer, second ed. 2011.
2. Taylor, R., Medvidovic, N., Dashofy, E., Software Architecture: Foundations, Theory, and Practice, 2010, Wiley.
3. Len Bass, Paul Clements, Rick Kazman, Software Architecture in Practice, 3rd edition, 2013.
4. David Patterson, Armando Fox, Engineering Long-Lasting Software: An Agile Approach Using SaaS and Cloud Computing, Alpha Ed.
5. Buschmann, Frank, Regine Meunier, Hans Rohnert, Peter Sornmerlad, and Michael Stal. 2001. Pattern-oriented system architecture, volume 1: A system of patterns. Hoboken, NJ: John Wiley & Sons. [POSA book]
6. Fowler Martin, Patterns of Enterprise Application Architecture, Addison-Wesley Professional, 2002.
7. Course materials published at https://users.utcluj.ro/~dinso/PS2017

8.2 Applications – Seminars/Laboratory/Project	Hours	Teaching methods	Notes
Revision exercises (OOP, UML, testing techniques)	2	Face-to-Face tutoring, additional materials	
Database connections and operations	2		
Architectural styles exercises	2		
Assignment 1 presentation and discussion	2		
Assignment 1 progress and discussion	2		
XML basics - exercises	2		
Design patterns exercises	2		
Assignment 2 presentation and discussion	2		
Assignment 2 progress and discussion	2		
Class design principles exercises	2		
Package design principles exercises	2		
Assignment 3 presentation and discussion	2		
Assignment 3 progress and discussion	2		
Assignments catch-up session	2		

Bibliography

Course materials published at https://users.utcluj.ro/~dinso/PS2017
Java tutorial - docs.oracle.com/javase/tutorial/
C# tutorial – msdn.microsoft.com

Se vor preciza, după caz: tematica seminariilor, lucrările de laborator, tematica și etapele proiectului.

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

ACM Curriculum compliant course

10. Evaluation

Activity type	Assessment criteria	Assessment methods	Weight in the final grade
Course	Ability to understand requirements, analyse alternative solutions and design an appropriate solution	Written exam	60%
Seminar			
Laboratory	Analyse requirements and alternative solutions, design an appropriate solution and implement it in either java or C#.	Periodic presentations of the required deliverables	40%
Project			
Minimum standard of performance: Grade calculus: 20% lab + 20% project + 60% final exam Conditions for participating in the final exam: Lab ≥ 5, Project ≥ 5 Conditions for promotion: final exam ≥ 5			

Course responsible
Prof.dr.eng. Mihaela Dinsoreanu

Head of department
Prof.dr.eng. Rodica Potolea

SYLLABUS

1. Data about the program of study

1.1 Institution	The Technical University of Cluj-Napoca
1.2 Faculty	Faculty of Automation and Computer Science
1.3 Department	Computer Science
1.4 Field of study	Computer Science and Information Technology
1.5 Cycle of study	Bachelor of Science
1.6 Program of study/Qualification	Computer science/ Engineer
1.7 Form of education	Full time
1.8 Subject code	42.

2. Data about the subject

2.1 Subject name	***Intelligent systems***				
2.2 Course responsible/lecturer	Prof. dr. eng. Leţia Ioan Alfred – Ioan.Alfred.Letia@cs.utcluj.ro				
2.3 Teachers in charge of seminars/ laboratory/ project	Assoc.prof. dr. eng. Razvan Slăvescu – Razvan.Slavescu@cs.utcluj.ro Assoc.prof. dr. eng. Anca Marginean – Anca.Marginean@cs.utcluj.ro				
2.4 Year of study	III	2.5 Semester	2	2.6 Type of assessment (E - exam, C - colloquium, V - verification)	E
2.7 Subject category	*DF – fundamentală, DD – în domeniu, DS – de specialitate, DC – complementară*				DS
	DI – Impusă, DOp – opțională, DFac – facultativă				DI

3. Estimated total time

3.1 Number of hours per week	4	of which:	Course	2	Seminars		Laboratory	2	Project	
3.2 Number of hours per semester	56	of which:	Course	28	Seminars		Laboratory	28	Project	

3.3 Individual study:	
(a) Manual, lecture material and notes, bibliography	18
(b) Supplementary study in the library, online and in the field	5
(c) Preparation for seminars/laboratory works, homework, reports, portfolios, essays	10
(d) Tutoring	6
(e) Exams and tests	9
(f) Other activities:	0

3.4 Total hours of individual study (suma (3.3(a)…3.3(f)))	48
3.5 Total hours per semester (3.2+3.4)	104
3.6 Number of credit points	4

4. Pre-requisites (where appropriate)

4.1 Curriculum	Logic Programming, Functional Programming
4.7 Competence	Fundamentals of Computer Programming

5. Requirements (where appropriate)

5.1. For the course	Projector, Computer
5.2. For the applications	Computers with Linux, Specific Software

6. Specific competence

6.1 Professional competences	**C6 –** Design of intelligent systems (4 credits) **C6.1 –** Describing the components of intelligent systems **C6.2 –** Usage of specific instruments of the domain for explaining and understanding the functioning of intelligent systems **C6.3 –** Application of principles and basic methods for the specification of solutions typical problems using intelligent systems **C6.4 –** Choosing criteria and methods for the evaluation of quality, performance and limits of intelligent systems **C6.5 –** Development and implementation of professional designs for intelligent

	systems
6.2 Cross competences	N/A

7. Discipline objective (as results from the *key competences gained*)

7.1 General objective	Knowledge of representation and reasoning of fundamental problems of intelligent systems
7.2 Specific objectives	Fundamental methods for basic representations in intelligent systems: uncertainty, learning, communication

8. Contents

8.1 Lectures	Hours	Teaching methods	Notes
Introduction.	2	Slides, Algorithms, Quality of solutions, Exceptions, Limits in the representation of the real world	
Uncertainty: inference using full joint distributions, Bayes' rule and its use.	2		
Probabilistic Reasoning: semantics of Bayesian networks, efficient representation, exact inference, approximate.	2		
Probabilistic Reasoning over Time: hidden Markov models, dynamic Bayesian networks.	2		
Making Simple Decisions: utility functions, decision networks, value of information.	2		
Making Complex Decisions: value iteration, policy iteration, partially observable MDPs, game theory.	2		
Learning from Observations: learning decision trees, ensemble learning.	2		
Knowledge in Learning: explanation-based, relevance information, inductive logic programming.	2		
Statistical Learning Methods: hidden variables, instance-based, neural networks, kernel machines.	2		
Reinforcement Learning.	2		
Association analysis: frequent itemset generation, rule generation, compact representation of frequent itemsets, alternative methods of generating frequent itemsets, FP-growth algorithm.	2		
Communication: syntactic analysis, semantic interpretation.	2		
Perception, representation and action in multi-agent systems.	2		
Overview on Intelligent Systems: Present and Future.	2		
Bibliography 1. Artificial Intelligence: A Modern Approach: Russell, Norvig, Prentice Hall, 2002 2. Tan, Steinbach, Kumar: Data Mining: Association Analysis, 2004			

8.2 Applications – Seminars/Laboratory/Project	Hours	Teaching methods	Notes
Introduction to the documentation for the assignment	2	Platform, Documentation, Testing, Examples, New examples	
Studying the documentation for the assignment	2		
Studying the design of the tool	2		
Practicing the exercises provided in the archive	2		
Understanding the main parts of the software	2		
Running the system by tracing at high level	2		
Mastering the running of the system and the examples provided	2		
Conceptual design of new examples	2		
Code for the new examples	2		
Testing and debugging the new cases	2		
Measuring the performance of the system	2		
Documenting the new scenarios	2		
Comparison of the differences between the cases developed and those provided	2		
Final evaluation of the exercises developed	2		

Bibliography 1. Various Intelligent Systems Tools from the WWW.

*Se vor preciza, după caz: tematica seminariilor, lucrările de laborator, tematica și etapele proiectului.

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

The course outline represents the most known and used one in the world methods for intelligent systems, continuously assessed by the research community in the world regarding its influence and use in software technology.

10. Evaluation

Activity type	Assessment criteria	Assessment methods	Weight in the final grade
Course	Problems and theoretical concepts	Written exam	80%
Seminar			
Laboratory	Usage of specific tools on the examples developed and tested by the students	Evaluation in the laboratory	20%
Project			
Minimum standard of performance: Representation of knowledge and its use in solving specific intelligent systems problems using specific tools. Grade calculus: 20% laborator + 80% examen final Conditions for participating in the final exam: Laborator ≥ 5 Conditions for promotion: grade ≥ 5			

Course responsible
Prof. dr. eng. Leţia Ioan Alfred

Head of department
Prof.dr.eng. Rodica Potolea

SYLLABUS

1. Data about the program of study

1.1 Institution	The Technical University of Cluj-Napoca
1.2 Faculty	Faculty of Automation and Computer Science
1.3 Department	Computer Science
1.4 Field of study	Computer Science and Information Technology
1.5 Cycle of study	Bachelor of Science
1.6 Program of study/Qualification	Computer science/ Engineer
1.7 Form of education	Full time
1.8 Subject code	43.

2. Data about the subject

2.1 Subject name	***Practical Placement***				
2.2 Course responsible/lecturer	Assoc. prof. dr. eng. Tiberiu Marita				
2.3 Teachers in charge of seminars/ laboratory/ project	Internship supervisors appointed by the faculty				
2.4 Year of study	III	2.5 Semester	2	2.6 Type of assessment (E - exam, C - colloquium, V - verification)	V
2.7 Subject category	*DF – fundamentală, DD – în domeniu, DS – de specialitate, DC – complementară*				DD
	DI – Impusă, DOp – opțională, DFac – facultativă				DI

3. Estimated total time

3.1 Number of hours per week	-	of which:	Course	-	Seminars	-	Laboratory	-	Project	-
3.2 Number of hours per semester	-	of which:	Course	-	Seminars	-	Laboratory	-	Project	-

3.3 Individual study:	
(a) Manual, lecture material and notes, bibliography	0
(b) Supplementary study in the library, online and in the field	40
(c) Preparation for seminars/laboratory works, homework, reports, portfolios, essays	10
(d) Tutoring	18
(e) Exams and tests	2
(f) Other activities:	170

3.4 Total hours of individual study (suma (3.3(a)…3.3(f)))	240
3.5 Total hours per semester (3.2+3.4)	240
3.6 Number of credit points	8

4. Pre-requisites (where appropriate)

4.1 Curriculum	N/A
4.8 Competence	N/A

5. Requirements (where appropriate)

5.1. For the course	N/A
5.2. For the applications	N/A

6. Specific competence

6.1 Professional competences	**C2** Designing hardware, software and communication components (2 credits) **C2.1** Describing the structure and functioning of computational, communication and software components and systems **C2.2** Explaining the role, interaction and operation of hardware, software and communication components **C2.3** Construction of hardware and software components of computing systems using design methods, languages, algorithms, data structures, protocols and technologies **C2.4** Metric based evaluation of functional and non-functional characteristics of

	computing systems **C2.5** Implementation of hardware, software and communication components **C3** Problems solving using specific Computer Science and Computer Engineering tools (2 credits) **C3.1** Identifying classes of problems and solving methods that are specific to computing systems **C3.2** Using interdisciplinary knowledge, solution patterns and tools, making experiments and interpreting their results **C3.3** Applying solution patterns using specific engineering tools and mehods **C3.4** Comparatively and experimentaly evaluation of the alternative solutions for performance optimization **C3.5** Developing and implementing informatic solutions for concrete problems **C5** Designing, managing the lifetime cycle, integrating and ensuring the integrity of hardware, software and communication systems (2 credits) **C5.1** Specifying the relevant criteria regarding the lifetime cycle, quality, security and computing system's interaction with the environment and human operator **C5.2** Using interdisciplinary knowledge for adapting an information system to application domain requirements **C5.3** Using fundamental principles and methods for security, reliability and usability assurance of computing systems **C5.4** Adequate utilization of quality, safety and security standards in information processing **C5.5** Creating a project including the problem's identification and analysis, its design and development, also proving an understanding of the basic quality requirements
6.2 Cross competences	**CT2** Identifying, describing and conducting processes in the projects management field, assuming different roles inside the team and clearly and concisely describing, verbally or in writing, in Romanian and in an international language, the results from the activity field. (2 credits)

7. Discipline objective (as results from the *key competences gained*)

7.1 General objective	Application of fundamental and applied knowledge gained in the projects development within a specialized company or research team (theme set by the project manager)
7.2 Specific objectives	Acquaintance and student involvement in every development stage of a hardware / software / communication project and connected aspects of design activities: - Analysis and documentation - Study and acquaintance with specific design tools - Design, implementation, testing and validation of the project - Preparation of documentations, technical reports - Team work and communication skills - Project management activities

8. Contents

8.1 Lectures	Hours	Teaching methods	Notes
-			
Bibliography -			
8.2 Applications – Seminars/Laboratory/Project	**Hours**	**Teaching methods**	**Notes**
• Study / documentation / training / • analysis of the potential methodologies and/or technologies • preparation of the project specifications • implementation and deployment of the hardware or software system • product testing and validation		N/A	

• product documenting			
Bibliography For the project development, the draft bibliography is the one recommended by the project leader from the company or by the research team at which the implementation is performed and the one resulted in the documenting phase.			

Se vor preciza, după caz: tematica seminariilor, lucrările de laborator, tematica și etapele proiectului.

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

This discipline provides education and training of the students at the workplace site, with benefits for both sides. Students are familiarized with the working and professional requirements needed to work in a company, and companies have the opportunity to shape students to facilitate their employment after graduation (to reduce training expenses / training). Also it aims to increase cohesion between academia and employment in a priority area in terms of national and European level in order to improve the skills of employees and to prepare and maintain them in the labor market in a particularly dynamic and competitive domain (mainly existing competition with Eastern European countries and Asia - India and China).

10. Evaluation

Activity type	Assessment criteria	Assessment methods	Weight in the final grade
Course			
Seminar			
Laboratory			
Project	Attendance (min 240 h), activity, tutor assessment	Colloquy	100%
Minimum standard of performance: Development of a hardware / software / communication engineering project.			

Course responsible
Assoc.prof.dr.eng. Tiberiu Marita

Head of department
Prof.dr.eng. Rodica Potolea

SYLLABUS

1. Data about the program of study

1.1 Institution	The Technical University of Cluj-Napoca
1.2 Faculty	Faculty of Automation and Computer Science
1.3 Department	Computer Science
1.4 Field of study	Computer Science and Information Technology
1.5 Cycle of study	Bachelor of Science
1.6 Program of study/Qualification	Computer science/ Engineer
1.7 Form of education	Full time
1.8 Subject code	111.

2. Data about the subject

2.1 Subject name	***Real time systems***				
2.2 Course responsible/lecturer	Prof. Tiberiu Letia; Tiberiu.Letia@aut.utcluj.ro				
2.3 Teachers in charge of seminars/ laboratory/ project	Prof. Tiberiu Letia; Tiberiu.Letia@aut.utcluj.ro				
2.4 Year of study	III	2.5 Semester	2	2.6 Type of assessment (E - exam, C - colloquium, V - verification)	C
2.7 Subject category	*DF – fundamentală, DD – în domeniu, DS – de specialitate, DC – complementară*				DS
	DI – Impusă, DOp – opțională, DFac – facultativă				DFac

3. Estimated total time

3.1 Number of hours per week	4	of which:	Course	2	Seminars		Laboratory	2	Project	
3.2 Number of hours per semester	56	of which:	Course	28	Seminars		Laboratory	28	Project	

3.3 Individual study:	
(a) Manual, lecture material and notes, bibliography	15
(b) Supplementary study in the library, online and in the field	15
(c) Preparation for seminars/laboratory works, homework, reports, portfolios, essays	15
(d) Tutoring	0
(e) Exams and tests	3
(f) Other activities:	0

3.4 Total hours of individual study (suma (3.3(a)...3.3(f)))	48
3.5 Total hours per semester (3.2+3.4)	104
3.6 Number of credit points	4

4. Pre-requisites (where appropriate)

4.1 Curriculum	Basic programming Software engineering Discrete event systems
4.9 Competence	Computer programming

5. Requirements (where appropriate)

5.1. For the course	N/A
5.2. For the applications	

6. Specific competence

6.1 Professional competences	C2 – Fundamental operation concepts from computer science information and communication technologies C5 Application development, algorithm implementations and control structures using project management principles, programming environment and microcontroller based technologies, DSP, PLC an embedded systems
6.2 Cross competences	N/A

7. Discipline objective (as results from the *key competences gained*)

7.1 General objective	
7.2 Specific objectives	

8. Contents

8.1 Lectures	Hours	Teaching methods	Notes
Real-Time Systems (RTS). Introduction to RTS	2	Interactive, multimedia	
Paradigms, basic definitions, RTS characteristics, real-time control, temporal parameters	2		
Specification of real-time applications (RTAs)	2		
Modeling of RTAs using Petri nets	2		
Design of RTAs with Unified Modeling Language	2		
Real-Time operating systems	2		
Interprocess communication	2		
Interrupt handling	2		
Concurrent programming in standard Java	2		
Implementation using Realtime Java	2		
Evaluation and measuring of execution times	2		
Scheduling (tests and verification)	2		
Verification and test of RT implementation	2		
Reliability of RTAs	2		

Bibliography

1. T. Leţia. Sisteme de timp-real. Editura Albastră (Microinformatica), ISBN 973-9443-49-4, 2001 (363 pag.).
2. T. Letia, A. Astilean. Sisteme cu evenimente discrete: modelare, analiză şi control. Editura Albastră (Microinformatica), Cluj-Napoca, ISBN. 973-9215-76-9, 1998 (228 pag.).
3. B. Bărbat, F.G. Filip. Informatică industrială. Ingineria programării în timp-real. Ed. Tehnică, Bucureşti, 1997.
4. J.E. Cooling. Software Design for Real-time Systems. International Thomson Computer Press, London, 1991.
5. Alan Burns, A. Wellings. Real-Time Systems and Programming Languages. Addison Wesley, 2001
6. A.M.K. Cheng. Real-Time Systems. Scheduling, Analysis and Verification, John Wiley and Sons, 2002
7. G. Buttazzo. Real-Time Systems. Predictable Scheduling and Applications. Springer, 2005.
8. Bruce Powel Douglass. Real-Time UML. Third Edition. Advances in The UML for Real-Time Systems. Ed. Addison-Wesley. 2007.
9. E.J.Brubo şi Greg Bollella. Real_Time Java Programming with Java RTS. Sun Micorsystems, 2009.
10. B.P. Douglass. Real Time UML Third Edition. Advances in the UML for Real-Time Systems. Addison-Wesley, 2007
11. A.M.K. Cheng. Real-Time Systems Scheduling, Analysis, and Verification. Ed. Wiley Interscience, JohnWiley and Sons, 2002.
12. G.C. Buttazzo. Hard Real-Time Computing. Predictable Scheduling Algorithms and Application. Second Edition. Ed. Springer. 2005.

8.2 Applications – Seminars/Laboratory/Project	Hours	Teaching methods	Notes
L1. Introduction – Tool and development environment	2	Interactive	
L2. Threads in Java SE – General concepts	2		
L3. Threads in Java SE – Classic synchronization mechanisms	2		
L4. Applications with threads in Java SE – Petri nets and Time Petri nets implementations and tests using classic synchronization mechanisms	2		
L5. Threads in Java SE – Package java.util.concurent - Part 1	2		
L6. Threads in Java SE – Package java.util.concurent - Part 2	2		
L7. Applications with threads in Java SE - Petri nets and Time Petri nets implementations and tests using synchronization mechanisms from the package java.util.concurent	2		
L8. Real-Time Java – Introductory notions	2		
L9. Clocks and timings in Real-Time Java	2		
L10. Real time threads	2		
L11. Applications with RT Java threads	2		
L12. Memory management in Real-Time Java	2		
L13. Compensatory activities	2		

L14. Final test -	2		

Bibliography

1. T. Leţia. Sisteme de timp-real. Editura Albastră (Microinformatica), ISBN 973-9443-49-4, 2001 (363 pag.).
2. T. Letia, A. Astilean. Sisteme cu evenimente discrete: modelare, analiză şi control. Editura Albastră (Microinformatica), Cluj-Napoca, ISBN. 973-9215-76-9, 1998 (228 pag.).
3. B. Bărbat, F.G. Filip. Informatică industrială. Ingineria programării în timp-real. Ed. Tehnică, Bucureşti, 1997.
4. J.E. Cooling. Software Design for Real-time Systems. International Thomson Computer Press, London, 1991.
5. Alan Burns, A. Wellings. Real-Time Systems and Programming Languages. Addison Wesley, 2001
6. A.M.K. Cheng. Real-Time Systems. Scheduling, Analysis and Verification, John Wiley and Sons, 2002
7. G. Buttazzo. Real-Time Systems. Predictable Scheduling and Applications. Springer, 2005.
8. Bruce Powel Douglass. Real-Time UML. Third Edition. Advances in The UML for Real-Time Systems. Ed. Addison-Wesley. 2007.
9. E.J.Brubo şi Greg Bollella. Real_Time Java Programming with Java RTS. Sun Micorsystems, 2009.
10. B.P. Douglass. Real Time UML Third Edition. Advances in the UML for Real-Time Systems. Addison-Wesley, 2007
11. A.M.K. Cheng. Real-Time Systems Scheduling, Analysis, and Verification. Ed. Wiley Interscience, JohnWiley and Sons, 2002.

Se vor preciza, după caz: tematica seminariilor, lucrările de laborator, tematica și etapele proiectului.

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

10. Evaluation

Activity type	Assessment criteria	Assessment methods	Weight in the final grade
Course	Midterm exam Final exam The grade for the midterm exam (M); The grade for the final exam (F)	written examination 2 hours written examination 3 hours	0.33 0.33
Seminar			
Laboratory	Laboratory verification (L)		0.33
Project			
Minimum standard of performance: N=0.33M+0.33F+ 0.33L Condition to obtain the credits : N≥5; M≥5; F≥5; L≥5			

Course responsible	Head of department
Prof.dr.eng. Tiberiu Letia	Prof.dr.eng. Rodica Potolea

SYLLABUS

1. Data about the program of study

1.1 Institution	The Technical University of Cluj-Napoca
1.2 Faculty	Faculty of Automation and Computer Science
1.3 Department	Computer Science
1.4 Field of study	Computer Science and Information Technology
1.5 Cycle of study	Bachelor of Science
1.6 Program of study/Qualification	Computer science/ Engineer
1.7 Form of education	Full time
1.8 Subject code	112.

2. Data about the subject

2.1 Subject name	***Managementul clasei de elevi***				
2.2 Course responsible/lecturer	Prof dr. Ing Bal Carmen				
2.3 Teachers in charge of seminars/ laboratory/ project	Prof dr. Ing Bal Carmen				
2.4 Year of study	III	2.5 Semester	2	2.6 Type of assessment (E - exam, C - colloquium, V - verification)	E
2.7 Subject category	*DF – fundamentală, DD – în domeniu, DS – de specialitate, DC – complementară*				DC
	DI – Impusă, DOp – opțională, DFac – facultativă				DFac

3. Estimated total time

3.1 Number of hours per week	2	of which:	Course	1	Seminars	1	Laboratory		Project
3.2 Number of hours per semester	28	of which:	Course	14	Seminars	14	Laboratory		Project

3.3 Individual study:	
(a) Manual, lecture material and notes, bibliography	10
(b) Supplementary study in the library, online and in the field	14
(c) Preparation for seminars/laboratory works, homework, reports, portfolios, essays	20
(d) Tutoring	3
(e) Exams and tests	3
(f) Other activities:	0

3.4 Total hours of individual study (suma (3.3(a)...3.3(f)))	50
3.5 Total hours per semester (3.2+3.4)	78
3.6 Number of credit points	3

4. Pre-requisites (where appropriate)

4.1 Curriculum	-
4.10Competence	-

5. Requirements (where appropriate)

5.1. For the course	• Participare activă
5.2. For the applications	• Lectura bibbliografiei recomandate • Documentare suplimentară • Elaborarea şi susţinerea prezentărilor planificate

6. Specific competence

6.1 Professional competences	C1: Proiectarea unor programe de instruire sau educaţionale adaptate pentru diverse niveluri de vârstă/pregătire şi diverse grupuri ţintă; C4 Abordarea managerială a grupului de şcolari, a procesului de învăţământ şi a activităţilor de învăţare/integrare socială specifice vârstei grupului ţintă C6 .Autoevaluarea şi ameliorarea continuă a practicilor profesionale şi a evoluţiei în carieră

6.2 Cross competences	CT2 Cooperarea eficienta în echipe de lucru profesionale, interdisciplinare, specifice desfasurarii proiectelor si programelor din domeniul stiintelor educatiei; CT4: Promovarea valorilor asociate realizării unui învăţământ de calitate, în conformitate cu politicile educaţionale interne şi în acord cu cele elaborate şi popularizate la nivel european, pe baza cunoaşterii specificităţii domeniului educaţional european şi a interculturalităţii; CT6 Aplicarea principiilor si a normelor de deontologie profesionala, fundamentate pe optiuni valorice explicite, specifice specialistului în stiintele educatiei;

7. Discipline objective (as results from the *key competences gained*)

7.1 General objective	• Să aplice tehnici eficiente de management al clasei de elevi, în cadrul diferitelor componente ale managementului clasei de elevi;
7.2 Specific objectives	• Să stabilească specificitatea abordării manageriale în procesul de învățământ; • Să analizeze componentele managementului clasei de elevi; • Să opereze cu conceptele specifice domeniului; • Să identifice situațiile de criză educațională încă din faza incipientă, ordonându-le şi clasificându-le în funcţie de specificitatea acestora; • Să determine soluţiile pertinente pentru diferitele situaţii de criză educaţională; • Să-şi perfecţioneze stilul managerial propriu.

8. Contents

8.1 Lectures	Hours	Teaching methods	Notes
1. Obiectul şi problematica managementului clasei de elevei. Conceptele de management general, educaţional, organizaţional – definire şi prezentare comparativă;	1	Curs interactiv: - expunerea; - prelegerea intensificată; - explicaţia; - conversaţia euristică; -problematizarea; - dezbaterea; - Jigsaw.	
2. Caracteristicile generale ale conducerii în sistemul de învăţământ. Principiile şi funcţiile managementului educaţional;	1		
3. Stiluri manageriale ale cadrelor didactice şi climatul şcolii;	1		
4. Clasa ca grup social. Relaţiile educaţionale;	1		
5. Utilitatea cunoaşterii clasei ca grup social;	1		
6. Managementul activităţilor didactice	1		
7. Managementul conflictului în clasa de elevi.	1		
Bibliography			

8.2 Applications – Seminars/Laboratory/Project	Hours	Teaching methods	Notes
1. Aspecte introductive: prezentarea obiectivelor disciplinei şi a competenţelor vizate, bibliografia, precizarea sarcinilor de seminar, distribuirea temelor şi referatelor	1	- exerciţiul; - studiul de caz; - eseul; - problematiza-rea; - dezbaterea; - jocul de rol	
2. Comunicarea la nivelul clasei: tipuri de comunicare, scheme de comunicare. Aplicaţii;	1		
3. Metode şi tehnici de cunoaştere a grupului şcolar: observaţia ştiinţifică	1		
4. Tehnica sociometrică, profilul psihosocial al grupului, autobiografia grupului	1		
5. Fişa de caracterizare psihosocială a clasei	1		
6. Managementul conflictului: studii de caz;	1		
7. Negocierea: tehnici de negociere – joc de rol.	1		

Bibliography

1. Băban, Adriana - *Consiliere educaţională,* Imprimeria Ardealul, Cluj-Napoca, 2001
2. Ciot Gabriela Melania – Managementul clasei de elevii, UTPRESS Cluj Napoca, 2006.
3. Ciascai, Liliana – *Managementul clasei de elevi. De la teorie la practică,* Ed. Casa Cărţii de Ştiinţă, Cluj-Napoca, 2007
4. Honţuş, Dumitru, Honțuș, Adelaida – *Managementul clasei de elevi*, Ed. Ceres, Bucureşti, 2008

5. Iucu, Romiţă B. – *Managementul clasei de elevi*, Polirom, Iaşi, 2006.
6. Lemeni, Gabriela., Miclea, Mircea - *Consiliere și orientare*, Ed. ASCR, Cluj-Napoca, 2004
7. Joiţa, Elena– *Management educaţional*, Polirom, Iaşi, 2000.
8. Niculescu, Rodica M. – *A învăţa să fii un bun manager*, Editura Inedit, Tulcea, 1994.
9. Orţan, Florica – *Management educaţional*, Editura Universităţii din Oradea, 2003.
10. Păun, Emil – *Şcoala - abordare sociopedagogică*, Polirom, Iaşi, 1999.
11. Rey, Bernard – *Faire la classe à l'école élémentaire*, ESF Editeur, 4e édition, Issy-les-Moulineaux, 2005.
12. Schulman Kolumbus, Elinor – *Didactică preşcolară*, Ediţia a II-a, V&I Integral, Bucureşti, 2000.
13. Stan, Emil – *Managementul clasei*, Aramis, Bucureşti, 2003.
14. Stan, Emil – *Profesorul între autoritate şi putere*, Teora, Bucureşti, 1999.
15. Ţoca, Ioan – *Management educaţional*, E.D.P., Bucureşti, 2002.
16. Cristea, G., Managementul lecţiei, Editura Didactică şi Pedagogică, R.A, Bucureşti, 2007;
17. Ezechil, L., Comunicarea educaţională în context şcolar, Bucureşti, E.D.P., 2002;
18. Iucu, R., Managementul clasei de elevi. Aplicaţii pentru gestionarea situaţiilor de criză educaţională, Editura Polirom, Iaşi, 2006;
19. Joita, E., Management educaţional : Profesorul-manager. Rolurişi metodologie, Editura Polirom, Iaşi, 2000
20. Alois Gherduţ, Managementgeneral şi strategic în educaţie, Ghid practic, Ed. Polirom, Iaşi, 2007.
21. Langa C. Tănase M., ionuţ B., Elemente de management educaţional, Editura Paradigme, 2003
22. Nicola, I., Microsociologia colectivului de elevi, Editura Didactică şi Pedagogică, Bucureşti, 1997;
23. Loca I., Management educaţional, Editura Didactică şi Pedagogică, R.A, Bucureşti, 2007
24. Ion Ovidiu Painişoară, Comunicare eficientă, Metode de interacţiune educaţională, ed. Polirom Iasi, 2008 ;
25. Potolea, D, Iucu, R., Neacşu, I., Pânişoară, O.,(coord.), Pregătirea psihopedagogică. Manual pentru definitivat şi gradul didactic II, Editura Polirom, Iaşi, 2008;
26. Sachelarie, O., Langa, C., Bulgaru,I., Probleme de sociologia educaţiei, Editura universităţii din Piteşti, 2002;
27. Schein, R. C., (1985), Organizational Culture and Leaderschip, Jossey – Bass, san Francisco ;
28. Skilbeck, M (1984), School Based Curriculum Develepment, harper and Row, Londra.
29. Zlate, M., Zlate, C., Cunoaşterea şi activarea grupurilor sociale, Editura Politică, Bucureşti, 1982.
30. Vlăsceanu, M., (1993) Psihologia organizaţiilor şi a conducerii, Ed. Paidela, Bucureşti;
31. Wallace, M., (1991), School – Centred management Training, Paul Champan Educational Publishing, Portsmouth;
32. Well, M. (1992), Le management strategique, Armand Colin, Paris.

**Se vor preciza, după caz: tematica seminariilor, lucrările de laborator, tematica și etapele proiectului.*

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

Competenţele dobândite la absolvirea acestui curs permit absolventului, indiferent de specializare, o gestionare mai eficientă a vieţii personale şi profesionale, respectiv o inserţie productivă pe piaţa forţei de muncă (prin cunoştinţele şi competenţele privind: managementul stresului, al timpului, cunoaşterea posibilităţilor personale şi profesionale reale, autodepăşire şi motivare, comunicare eficientă ş.a.).

10. Evaluation

Activity type	Assessment criteria	Assessment methods	Weight in the final grade
Course	Volumul şi corectitudinea cunoştinţelor	Lucrare scrisă	40
	Rigoarea ştiinţifică a limbajului	Lucrare scrisă	10
	Organizarea conţinutului	Lucrare scrisă	10
	Originalitatea	Lucrare scrisă	10
Seminar	Susţinerea unui referat	Fişă de evaluare seminar	20
	Participare activă la seminarii	Fişă de evaluare seminar	10
Laboratory			
Project			
Minimum standard of performance: 50% rezultat după însumarea punctajelor ponderate			

Course responsible
Prof.dr.eng. Carmen Bal

Head of department
Prof.dr.eng. Rodica Potolea