SYLLABUS

1. Data about the program of study

1.1	Institution	The Technical University of Cluj-Napoca
1.2	Faculty	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	23.

2. Data about the subject

2.1	Subject name					Systems Theory					
2.2	Subject area					Computer Science and Information Technology					
2.3	Course responsible/lecturer					Assoc. prof. dr. eng. Paula Raica – Paula.Raica@aut.utcluj.ro					
2.4	Teachers in charge of applications					Sl.dr.ing. Ionut Muntean – Ionut.Muntean@aut.utcluj.ro Sl.dr.ing. Lucian Busoniu – Lucian.Busoniu@aut.utcluj.ro, Sl.dr.ing. Cosmin Marcu – Cosmin.Marcu@aut.utcluj.ro					
2.5	Year of study	II	2.6	Semester	4	2.7	Assessment	exam	2.8	Subject category	DID/OB

3. Estimated total time

Sem.	Subject name	Lecture	Applications			Lecture	Applications			Individual study	TOTAL	Credit
		[hours / week.]				[hours / semester]						
			S	L	P		S	L	P			
4	**Systems Theory**	**2**	**-**	**2**	**-**	**28**	**-**	**28**	**-**	**48**	**104**	**4**

3.1	Number of hours per week	4	3.2	of which, course	2	3.3	applications	2
3.4	Total hours in the teaching plan	56	3.5	of which, course	28	3.6	applications	28

Individual study	Hours
Manual, lecture material and notes, bibliography	20
Supplementary study in the library, online and in the field	5
Preparation for seminars/laboratory works, homework, reports, portfolios, essays	20
Tutoring	
Exams and tests	3
Other activities	

3.7	Total hours of individual study	48
3.8	Total hours per semester	104
3.9	Number of credit points	4

4. Pre-requisites (where appropriate)

4.1	Curriculum	Mathematical Analysis_II (Integral calculus and differential equations, Linear algebra
4.2	Competence	Differential equations, complex numbers, Laplace transform, linear algebra

5. Requirements (where appropriate)

5.1	For the course	N/A
5.2	For the applications	Reading and understanding of the lecture notes.

6. Specific competences

Professional competences	**C1** – Operating with basic Mathematical, Engineering and Computer Science concepts (4 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 **C1.4** – 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
Cross competences	N/A

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

7.1	General objective	The general objective of the course is to introduce the fundamental principles of linear system modeling, analysis and feedback control and to evaluate feedback control systems with desired behavior.
7.2	Specific objectives	The specific objectives are to acquire the knowledge and techniques related to: - mathematical system modeling (differential equations, input-output representation as transfer functions, block diagrams) for simple applications - linear system analysis (assessment of stability and performance properties of linear systems) in time and frequency domains - design of feedback controllers such as PID, lead and lag compensators for linear systems using s-domain techniques - linear sampled-data system representation and analysis

8. Contents

8.1. Lecture (syllabus)		Teaching methods	Notes
1	Introduction to system theory and control engineering	Lecture, visual presentations, demonstrations	N/A
2	Mathematical models of systems. Transfer functions and system response		
3	Block diagram models. Block diagram reduction. MIMO systems		
4	Analysis of linear continuous systems. 1st and 2nd order systems. Steady-state error.		
5	Higher order systems. Stability of linear continuous systems		
6	System analysis using root locus		
7	Frequency response. Bode diagrams.		
8	Frequency response. Stability in the frequency domain. Applications		
9	PID – the basic technique for feedback control;		
10	Controller design using root locus. Lead-lag compensators		
11	Controller design using root locus. Applications		
12	Sampled-data systems		
13	Digital control systems		
14	Sampled data and digital control systems. Applications		

Bibliography

1. R. C. Dorf, R. Bishop, "Modern Control Systems", Addison-Wesley, 2004;
2. K. Ogata , "Modern Control Engineering", Prentice Hall, 1990.
3. K. Dutton, S. Thompson, B. Barraclough, "The Art of Control Engineering", Addison-Wesley, 1997
4. William S. Levine (editor), "The Control Handbook", CRC Press and IEEE Press, 1996
5. Lecture notes available on the course webpage: http://rrg.utcluj.ro/ts

8.2. Applications (Seminars, Laboratory, Projects)		Teaching methods	Notes
1	Introduction to Matlab. Simulation of dynamical systems	Class discussion, Supervised exercise solving using Matlab Miniprojects –	4 hours
2	Linear approximation of differential equations. Transfer functions. System response.		4 hours
3	Block diagram models. 1st and 2nd order system analysis. Steady-state error		4 hours
4	System stability. Root locus		4 hours

5	Frequency response. Bode diagrams	individul student reports	4 hours
6	PID. Lead-lag compensation		4 hours
7	Sampled-data systems		4 hours

Bibliography

1. Paula Raica, "Control Engineering. Exercises", Editura Mediamira, 2001
2. Lecture notes available on the course webpage: http://rrg.utcluj.ro/ts

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

The course content combines theoretical knowledge with applications and focuses on the formulation and solution of specific problems that may occur in various engineering fields. Application of the control theory concepts are specific to most of the engineering disciplines. The course level is introductory and the intent is to motivate and prepare students for further study in related areas and to conduct projects in real-life applications.

10. Evaluation

Activity type	10.1	Assessment criteria	10.2	Assessment methods	10.3	Weight in the final grade
Course		Ability to solve exercises related to linear system modeling and analysis		Midterm exam		40%
		Ability to solve exercises related to system design and analysis of sampled-data systems		Final exam		60%
Applications		Answer simple questions from the topic of the lab applications		Lab tests (optional)		20%
		Submitting and defending a miniproject on a given subject		Individual student report (optional)		20%
10.4 Minimum standard of performance						
Solution of simple exercises applying the knowledge and techniques presented in the course						

Course responsible
Conf.dr.ing. Paula Raica

Head of department
Prof.dr.ing. Rodica Potolea

SYLLABUS

1. Data about the program of study

1.1	Institution	The Technical University of Cluj-Napoca
1.2	Faculty	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	24.

2. Data about the subject

2.1	Subject name					Computer Architecture					
2.2	Subject area					Computer Science and Information Technology					
2.3	Course responsible/lecturer					As.dr.eng. Mihai Negru – Mihai.Negru@cs.utcluj.ro					
2.4	Teachers in charge of applications					Conf.dr. eng. Florin Oniga, As.dr.eng. Mihai Negru, { Florin.Oniga, Mihai.Negru }@cs.utcluj.ro					
2.5	Year of study	II	2.6	Semester	4	2.7	Assessment	exam	2.8	Subject category	DID/OB

3. Estimated total time

Sem.	Subject name	Lecture	Applications			Lecture	Applications			Individual study	TOTAL	Credit
		[hours / week.]				[hours / semester]						
			S	L	P		S	L	P			
4	**Computer Architecture**	**2**	**-**	**2**	**-**	**28**	**-**	**28**	**-**	**74**	**130**	**5**

3.1	Number of hours per week	4	3.2	of which, course	2	3.3	applications	2
3.4	Total hours in the teaching plan	56	3.5	of which, course	28	3.6	applications	28

Individual study	Hours
Manual, lecture material and notes, bibliography	28
Supplementary study in the library, online and in the field	14
Preparation for seminars/laboratory works, homework, reports, portfolios, essays	28
Tutoring	0
Exams and tests	4
Other activities	0

3.7	Total hours of individual study	74
3.8	Total hours per semester	130
3.9	Number of credit points	5

4. Pre-requisites (where appropriate)

4.1	Curriculum	Logic design, Digital system design (VHDL)
4.2	Competence	Ability to design digital circuits and to implement them in VHDL

5. Requirements (where appropriate)

5.1	For the course	blackboard, video projector, laptop
5.2	For the applications	desktop/laptop computer, Xilinx ISE, FPGA development boards

6. Specific competences

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
Cross competences	N/A

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

7.1	General objective	Knowing and understanding the concepts of organization and functioning for central processing units, memories, input/output, and using these concepts for design.
7.2	Specific objectives	• Applying methods for representation and design at system level for digital circuits • Instruction Set Architecture (ISA) specification • Writing simple programs in assembly languages and machine code • Specification, design, implementation, and testing of Central Processing Units (CPU) – micro architecture – data path – command units • Understanding memory organization and I/O operations • Understanding modern trends in computer architectures

8. Contents

8.1. Lecture (syllabus)		Teaching methods	Notes
1	Introduction	Oral presentation backed up by multimedia equipment, interactive communication, blackboard problem solving	
2	High-Level Synthesis		
3	Instruction Set Architecture (ISA)		
4	CPU Design - Single Cycle CPU		
5	Computer Arithmetic and Simple Arithmetic Logic Units		
6	CPU Design - Multi Cycle CPU Data path		
7	CPU Design - Multi Cycle CPU Control		
8	CPU Design – Pipelined CPU		
9	Advanced Pipelining – Static and Dynamic Scheduling of the Execution		
10	Branch Prediction		
11	Superscalar Architectures		
12	Memory		
13	I/O and Interconnection Structures		
14	Problem solving		

Bibliography

1. D. A. Patterson, J. L. Hennessy, "Computer Organization and Design: The Hardware/Software Interface",5th edition, ed. Morgan–Kaufmann, 2013.
2. D. A. Patterson and J. L. Hennessy, "Computer Organization and Design: A Quantitative Approach",5th edition, ed. Morgan-Kaufmann, 2011.
3. Vincent P. Heuring, et al., "Computer Systems Design and Architecture", Addison-Wesley, USA, 1997.
4. A. Tanenbaum, "Structured Computer Organization", Prentice Hall, USA, 1999.
5. MIPS32 Architecture for Programmers, Volume I: "Introduction to the MIPS 32™ Architecture".
6. MIPS32 Architecture for Programmers, Volume II: "The MIPS 32™ Instruction Set".

Online bibliography
M. Negru, F. Oniga, S. Nedevschi, Lecture slides http://users.utcluj.ro/~negrum

8.2. Applications (Laboratory)		Teaching methods	Notes
1	Introduction in the Xilinx ISE environment and the FPGA development board	Blackboard quick overview of key issues, exercises, experimenting with FPGA development boards with specialized IDEs for circuit design and implementation (Xilinx ISE)	
2	Design and Implementation of Combinational CPU Components		
3	Design and Implementation of Sequential CPU Components		
4	Design of a Single Cycle CPU 1 (MIPS)		
5	Design of a Single Cycle CPU 2 (MIPS)		
6	Design of a Single Cycle CPU 3 (MIPS)		
7	Design of a Single Cycle CPU 4 (MIPS)		
8	Midterm practical evaluation on the FPGA board		
9	Pipelined CPU Design		
10	Pipelined CPU Design		
11	Pipelined CPU Design		
12	Pipelined CPU interfacing		
13	Practical evaluation of the pipelined CPU on the FPGA board		
14	Final Tests and Evaluation		
Bibliography ***Online bibliography*** 1. M. Negru, F. Oniga, S. Nedevschi, Laboratory guide http://users.utcluj.ro/~negrum			

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

Computer Architecture is one of the fundamental subjects of the Computer Science and Information Technology field. It combines fundamental and practical aspects used for digital circuits design and implementation. The content of this subject is harmonized with the specific curricula of other national and international universities, and is evaluated by the Romanian government agencies (CNEAA and ARACIS). The practical aspects involve getting familiar with and using development products and tools provided by companies from Romania, Europe, and USA (ex. Xilinx, Digilent).

10. Evaluation

Activity type	10.1	Assessment criteria	10.2	Assessment methods	10.3	Weight in the final grade
Course		Testing the theoretical knowledge mainly through the ability problem solving		Written exam		50 %
Applications		Practical ability to solve and implement specific problems related to processor design, presence and activity		Lab exam, periodical assessment of results		50 %
10.4 Minimum standard of performance						
Knowing the fundamental theory of the subject, the ability to design and implement a processor with a reduced set of instructions						

Course responsible
As.dr. eng. Mihai Negru

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	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	27.

2. Data about the subject

2.1	Subject name					Operating Systems					
2.2	Subject area					Computer Science and Information Technology					
2.3	Course responsible/lecturer					Lect. dr. eng. Adrian Coleşa – adrian.colesa@cs.utcluj.ro					
2.4	Teachers in charge of applications					Lect. dr. eng. Adrian Coleşa – adrian.colesa@cs.utcluj.ro Eng. Gheorghe Hajmasan – ghajmasan@bitdefender.com Eng. Andrei Lutas – vlutas@bitdefender.com Eng. Radu Ciocas – rciocas@bitdefender.com					
2.5	Year of study	II	2.6	Semester	4	2.7	Assessment	exam	2.8	Subject category	DID/OB

3. Estimated total time

Sem.	Subject name	Lecture	Applications			Lecture	Applications			Individual study	TOTAL	Credit
		[hours / week.]				[hours / semester]						
			S	L	P		S	L	P			
4	**Operating Systems**	**2**	**-**	**2**	**-**	**28**	**-**	**28**	**-**	**74**	**130**	**5**

3.1	Number of hours per week	4	3.2	of which, course	2	3.3	applications	2
3.4	Total hours in the teaching plan	56	3.5	of which, course	28	3.6	applications	28

Individual study	Hours
Manual, lecture material and notes, bibliography	30
Supplementary study in the library, online and in the field	10
Preparation for seminars/laboratory works, homework, reports, portfolios, essays	28
Tutoring	2
Exams and tests	4
Other activities	0

3.7	Total hours of individual study	74
3.8	Total hours per semester	130
3.9	Number of credit points	5

4. Pre-requisites (where appropriate)

4.1	Curriculum	Computer Programming, Data Structures and Algorithms
4.2	Competence	C programming

5. Requirements (where appropriate)

5.1	For the course	Students must have minimum 9 classes attended to be allowed to take the exam
5.2	For the applications	Students must have minimum 11 classes attended to be allowed to take the exam

6. Specific competences

Professional competences	**C3:** Problems solving using specific Computer Science and Computer Engineering tools (3 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 methods • **C3.4**: Evaluating, comparatively and experimentally, the available alternative solutions for performance optimization • **C3.5**: Developing and implementing informatic solutions for concrete problems **C4**: Improving the performances of the hardware, software and communication systems (2 credits) • **C4.1**: Identifying and describing the defining elements of the performances of the hardware, software and communication systems • **C4.2**: Explaining the interaction of the factors that determine the performances of the hardware, software and communication systems • **C4.3**: Applying the fundamental methods and principles for increasing the performances of the hardware, software and communication systems • **C4.4**: Choosing the criteria and evaluation methods of the performances of the hardware, software and communication systems • **C4.5**: Developing professional solutions for hardware, software and communication systems based on performance optimization
Cross competences	N/A

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

7.1	General objective	Have a clear understanding of what an OS is, its role and general functionality and be able to use most of the OS system calls.
7.2	Specific objectives	Have knowledge and understand the general structure and functionality of an OS. Understand the specific functionality of the most important OS components, like shell, process manager, file system, memory manager, security manager. Understand the functionality of main synchronization mechanisms and be able to use them to solve real synchronization problems. Be able to write C programs to use an OS (Linux and Windows) system calls.

8. Contents

8.1. Lecture (syllabus)		Teaching methods	Notes
1	**Introduction and basic concepts**. OS's definition, role, evolution, components, main concepts (file, process, system calls). Basic hardware aspects: CPU, user and kernel mode, memory layers, I/O devices. Basic OS structure.	(1) lecture presentation based on beamer presentation; (2) interactions with students: ask their opinion relative to the presented subject; (3) give each class a short evaluation test; let students discuss and argue each other their solution; give them the good solution and let them evaluate their own one; (4) propose 2-3 interesting study cases of OSes to be prepared and presented by students; (5) students are invited to collaborate in	
2	**The Shell (Command Interpreter)**. Definition, role, functionality, simple and complex commands. Standard input and output redirection.		
3	**File systems (1)**. User Perspective. File and directory concept from the user point of view (definition, role, characteristics, operations).		
4	**File systems (2)**. Windows and Linux File Systems. Permission rights and system calls.		
5	**File systems (3)**. Implementation aspects. Implementation strategies overview, space management and related problems, hard and symbolic links.		
6	**Process management**. Process model: definition, role, characteristics. Linux and Windows process management system calls.		
7	**Thread management**. Thread model: user vs. kernel threads, implementation problems, usage, performance aspects. Basic scheduling algorithms (FIFO, SJF, Priority-based). Linux and Windows process thread system calls.		
8	**Process synchronization (1)**. Theoretical aspects. Context, definition, synchronization mechanisms, techniques and problems (locks, semaphores, monitors, mutual exclusion, starvation, deadlock).		
9	**Process synchronization (2)**. Classical synchronization patterns: producer/consumer, readers/writers, rendez-vous, barrier, dining philosopher, sleeping barber. Similarities between different synchronization mechanisms.		

10	**Inter-process communication**. Pipe files, shared memory, message queues, signals.	research projects.	
11	**Memory management (1)**. Context, definition, binding, basic techniques, space management, addresses translation, swapping.		
12	**Memory management (2)**. Paging and segmentation.		
13	**I/O Devices Management**. Principles, disks, clocks, character-oriented terminals.		
14	**Security aspects**. Security policies and mechanisms. Basic program's vulnerabilities (buffer overflow).		

Bibliography

1. Andrew Tanenbaum. *Modern Operating System*, 2nd Edition, Prentice-Hall, 2005, ISBN 0-13-092641-8.

8.2. Applications (Laboratory)		Teaching methods	Notes
1	Linux File System	(1) students are presented a very brief overview of the most important and difficult aspects of the working subject ; (2) students are given at the beginning of each class a short evaluation quiz; (3) students are given a hands-on tutorial to practice with working subject's aspects and to solve problems (4) students are given challenging problems for extra credit;	
2	Linux Commands		
3	Linux Shell Scripts		
4	Linux System Calls for File Access		
5	Linux System Calls for File and Directory Manipulation		
6	Windows File System (NTFS)		
7	Linux Processes		
8	Linux Threads		
9	Windows Processes and Threads		
10	Linux Semaphores		
11	Linux Locks and Condition Variables		
12	Linux pipes		
13	Memory management.		
14	Security aspects. Buffer overflow detection and correction.		

Bibliography

Lecture slides and laboratory text and support at http://os.obs.utcluj.ro/moodle

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

OS knowledge is a fundamental requirement in the CS field. We follow the CAM curricula guide. We also consult IT companies about their practical expecations regarding OS knowledge and adapt accordingly our course contents. In this sense, Linux and Windows are the most used Oses.

10. Evaluation

Activity type	10.1	Assessment criteria	10.2	Assessment methods	10.3	Weight in the final grade
Course		Check if students: understand fundamental OS concepts and are able to recognize an OS-related problem and find solution to it.		Small problem-like subject requiring students to apply the theoretical OS aspects to give a solution to that problem.		0.67
Applications		Check if students are able to write C programs to use different OS provided system calls to solve practical problems.		Quiz tests. Programming problems, whose solution has to be implemented in C and run on computers.		0.33
10.4 Minimum standard of performance.						
Know the basic system calls, understand their functionality and be able to use them.						

Course responsible
Lect. dr. eng. Adrian Coleşa

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	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	28.

2. Data about the subject

2.1	Subject name					Elements of Computer Assisted Graphics					
2.2	Subject area					Computer Science and Information Technology					
2.3	Course responsible/lecturer					Prof. dr. eng. Gorgan Dorian – dorian.gorgan@cs.utcluj.ro					
2.4	Teachers in charge of applications					As. drd. eng. Melenti Cornelia, S.l. dr. eng. Bacu Victor, {cornelia.melenti, victor.bacu}@cs.utcluj.ro					
2.5	Year of study	II	2.6	Semester	4	2.7	Assessment	exam	2.8	Subject category	DF/OB

3. Estimated total time

Sem.	Subject name	Lecture	Applications			Lecture	Applications			Individual study	TOTAL	Credit
		[hours / week.]				[hours / semester]						
			S	L	P		S	L	P			
4	**Elements of Computer Assisted Graphics**	**2**	**-**	**2**	**-**	**28**	**-**	**28**	**-**	**48**	**104**	**4**

3.1	Number of hours per week	4	3.2	of which, course	2	3.3	applications	2
3.4	Total hours in the teaching plan	56	3.5	of which, course	28	3.6	applications	28

Individual study	Hours
Manual, lecture material and notes, bibliography	20
Supplementary study in the library, online and in the field	6
Preparation for seminars/laboratory works, homework, reports, portfolios, essays	10
Tutoring	3
Exams and tests	9
Other activities	0

3.7	Total hours of individual study	48
3.8	Total hours per semester	104
3.9	Number of credit points	4

4. Pre-requisites (where appropriate)

4.1	Curriculum	Computer programming (C language)
4.2	Competence	Applications development in C programming language

5. Requirements (where appropriate)

5.1	For the course	Projector, computer
5.2	For the applications	Laboratory attendance is mandatory Study of laboratory materials from the server

6. Specific competences

Professional competences	**C3 –** Problems solving using specific Computer Science and Computer Engineering tools (4 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 –** Evaluating, comparatively and experimentally, the available alternative solutions for performance optimization **C3.5 –** Developing and implementing informatic solutions for concrete problems
Cross competences	N/A

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

7.1	General objective	Learning about the architecture of a graphic system, the study of the graphic pipeline, the study of 2D graphic algorithms
7.2	Specific objectives	1. Creation of the graphical model of a scene of objects 2. Implementation of the basic algorithms that form the core of a graphic system 3. Development of graphic applications in a high-level programming language (C, C++) 4. Implementation of the main phases of the graphic transformation pipeline

8. Contents

8.1. Lecture (syllabus)		Teaching methods	Notes
1	Introduction. History. Examples	New multimedia teaching approaches will be used in classes. The course is interactive and includes demonstrations that exemplify graphical methods and algorithms.	During the semester and before each exam there are a few preparation hours planned.
2	Graphics systems – architecture, standards		
3	Graphics devices – logic and physics devices, input, output and interactive devices		
4	Graphics transformations pipeline – 2D and 3D transformations. Matrix operators		
5	Mathematics in computer graphics		
6	Lines scan conversion algorithms		
7	Circles scan conversion algorithms		
8	Polygons scan conversion algorithms		
9	Clipping algorithms – point, line, polygon and text		
10	Projections and viewing transformations		
11	Photorealistic presentation of 3D objects – concepts, algorithms, examples		
12	Color models – color perception, color space and standards, color in software design		
13	Graphics formats – vector and raster formats, data compression , Web technologies		
14	Graphics pattern grammars		

Bibliography

7. Foley J.D., van Dam, A., Feiner, S.K., Hughes, J.F., "*Computer Graphics. Principles and Practice*". Addison-Wesley Pblishing Comp., 1992.
8. Watt A., "*3D Computer Graphics*". Addison-Wesley, 1998.

In virtual libarry

1. Course resurses, http://cgis.utcluj.ro/didactic

8.2. Applications (Laboratory)		Teaching methods	Notes
1	Basic graphics application structure	Documentation and examples will be available to the students, prior to the laboratory classes, on a dedicated server.	Each student will have to develop a specific project based on the knowledge
2	Output and input operations in graphics window		
3	Inputs by keyboard, mouse and timer		
4	Menu, icon, cursor and bitmap resources – Part 1		
5	Menu, icon, cursor and bitmap resources – Part 2		
6	Coordinate systems. Viewing transformations		
7	2D clipping. Cohen-Sutherland Algorithm		

8	Projections. 2D and 3D transformations	The students will work independently but will also be assisted by the teacher.	acquired at the laboratory hours.
9	Lines scan conversion. Bresenham method		
10	Polygon clipping. Sutherland-Hodgman Algorithm		
11	Polygon clipping. Weiler Clipping Algorithm		
12	Photorealistic presentations		
13	Color computation		
14	Assessment		

Bibliography

In virtual libarry

1. Course and practical works, http://cgis.utcluj.ro

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

This discipline is integrated into the Computers and Information Technology domain. The content is classic, yet modern, and introduces to students the fundamentals of graphic systems and 2D algorithms. The content of this discipline has been aligned with the information presented in similar disciplines from other major universities and companies from Romania, Europe and USA and has been evaluated by the authorized Romanian governmental agencies (CNEAA and ARACIS).

10. Evaluation

Activity type	10.1	Assessment criteria	10.2	Assessment methods	10.3	Weight in the final grade
Course		The written exam tests the understanding of the information presented in classes and the ability to apply this knowledge. The activity in class evaluates the active involvement of the students in the teaching process and their participation to the discussions, debates and other class activities during the entire semester.		Evaluation is performed through written exam (E) and classes activity (AC)		50% (E) 10% (AC)
Applications		Laboratory assessment evaluates the practical abilities obtained by the students. Through homework assignments the students have the opportunity to develop their skill in applying the notions, concepts and methods presented in class.		Evaluation is performed through written exam.		40%
10.4 Minimum standard of performance						
Final mark: N=0,5*E+0,4*[(C+T)/2]+0.1*AC Graduation requirement: N≥5;						

Course responsible
Prof. dr. eng. Dorian Gorgan

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	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	29.

2. Data about the subject

2.1	Subject name					Foreign Language II (English, French, German - Technical documents elaboration)					
2.2	Subject area					Computer Science and Information Technology					
2.3	Course responsible/lecturer					Lect. dr. Sonia Munteanu Sonia.Munteanu@lang.utcluj.ro					
2.4	Teachers in charge of applications					-					
2.5	Year of study	II	2.6	Semester	4	2.7	Assessment	Colloquium	2.8	Subject category	DC/OB

3. Estimated total time

Sem	Subject name	Lecture	Applications			Lecture	Applications			Individual study	TOTAL	Credit
		[hours / week.]				[hours / semester]						
			S	L	P		S	L	P			
4	**Foreign Language II (English, French, German - Technical documents elaboration)**	**2**	**-**	**-**	**-**	**28**	**-**	**-**	**-**	**24**	**52**	**2**

3.1	Number of hours per week	2	3.2	of which, course	2	3.3	applications	-
3.4	Total hours in the teaching plan	28	3.5	of which, course	28	3.6	applications	-

Individual study	Hours
Manual, lecture material and notes, bibliography	6
Supplementary study in the library, online and in the field	8
Preparation for seminars/laboratory works, homework, reports, portfolios, essays	6
Tutoring	0
Exams and tests	4
Other activities	

3.7	Total hours of individual study	24
3.8	Total hours per semester	52
3.9	Number of credit points	2

4. Pre-requisites (where appropriate)

4.1	Curriculum	B1 according to the Common European Framework for Languages
4.2	Competence	Continuous training

5. Requirements (where appropriate)

5.1	For the course	Subjects of semester 1, year 2
5.2	For the applications	

6. Specific competences

Professional competences	N/A
Cross competences	**CT3** – Demonstrating the spirit of initiative and action for updating professional, economical and organizational culture knowledge (2 credits)

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

7.1	General objective	Development of integrated skills in an engineering professional context
7.2	Specific objectives	- Mastering documenting strategies, information processing; writing according to discourse patterns in specific purposes contexts; - Acquiring strategies for handling difficult written text on a variety of science related topics; - Comprehension and production of discipline appropriate text and genre. - Use of lexical and grammar structures at B1/B2 language competence levels, according to CEFL.

8. Contents

8.1. Lecture (syllabus)		Teaching methods	Notes
1	Hierarchical structure of grammar. Natural language processing; morphology, syntax, discourse. Language knowledge in technology development for language processing and artificial intelligence.	lecture, conversation, elicitation, practical application of knowledge, assignment discussion	
2	Student's research on NLP and AI topics which involve knowledge about language. Assignment discussion.		
3	Word structure: inflected and derivated words. Derivation as a means of creating technical vocabulary.		
4	Phrases: noun headed phrases, verb headed phrases, adjective headed phrases, preposition headed phrases.		
5	Simple and complex sentences. Frequently used phrase/sentence structures in technical texts: coordination and subordination in finite and non-finite clauses.		
6	Cohesion and coherence in discourse. Readibility of technical texts: syntactic parallelism, sentence rephrasal, nominalization, lexical choice		
7	Structure of information in paragraphs: general-particular patterns, theme-rheme, hypothesis and validation.		
8	Mid term evaluation..		
9	The informative function of science discourse: information structure, impersonal expression, nominalized theme.		
10	Functional and rhetorical organization of written science discourse: genres (textbooks, journal articles and scientific posters)..		
11	Information organization in scientific posters: functional and structural patterns. Design options and selection of information – from the journal article to the poster.		
12	Formulaic language in science discourse: multifunctional lexical bundles. Interpersonal function of science discourse: hedges, boosters and author mention in science discourse.		
13	Disciplinary variation in science discourse: professional communities, discourse communities. Selecting from language resources according to disciplinary practices.		
14	Test.		
Bibliography			

1.Munteanu, S.-C (2013) *Academic English for Science and Engineering*. Cluj-Napoca: Casa Cartii de Stiinta. ISBN 978-606-17-0398-2.
2.Swales John M. & Christine B. Feak (2001) *Academic Writing For Graduate Students - Essential Tasks And Skills*, Ann Arbor: The University Of Michigan Press.
3. Hyland Ken (2006) *English For Academic Purposes - An Advanced Resource Book*, London: Routledge

8.2. Applications (Seminars, Laboratory, Projects)		Teaching methods	Notes
1	-		
Bibliography -			

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

Mastering a foreign language will support students in a more flexible integration in the labour market, and have improved personal development. The introduction in the language for specific purposes will facilitate reading more documents in the field of study.

10. Evaluation

Activity type	10.1	Assessment criteria	10.2	Assessment methods	10.3	Weight in the final grade
Course		Assessment and homework completion in due time; Ability to comprehend below and above sentence syntactic and morphologic structures specific to science discourse; to read from sources, to comprehend complex text (journal articles, textbooks); write and present a poster.		Oral presentation of homework and assignment tasks; Two written tests or one test and science poster.		assignment/ homework = 20% mid-term test = 30% poster/final test = 40% poster presentation =10% total = 100%
Applications						
10.4 Minimum standard of performance						
Minimum 60% of the final test, regarding language, lexical and discourse structures used in the technical discourse, linking words, verbs in impersonal moods, nominal groups, revision and correction of written texts. Assignment completion, minimum 50% of the mid term evaluation.						

Course responsible
Lect. dr. Sonia Munteanu

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	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	105.

2. Data about the subject

2.1	Subject name					Mechanics					
2.2	Subject area					Computer Science and Information Technology					
2.3	Course responsible/lecturer					Prof. dr. eng. Vistrian Măties Email: vistrian.maties@mdm.utcluj.ro; vistrian.maties@yahoo.com					
2.4	Teachers in charge of applications					Dr. eng. Sergiu-Dan Stan					
2.5	Year of study	II	2.6	Semester	4	2.7	Assessment	Colloquium	2.8	Subject category	FAC

3. Estimated total time

Sem.	Subject name	Lecture	Applications			Lecture	Applications			Individual study	TOTAL	Credit
		[hours / week.]				[hours / semester]						
			S	L	P		S	L	P			
4	**Mechanics**	**2**	**2**	**-**	**-**	**28**	**28**	**-**	**-**	**48**	**104**	**4**

3.1	Number of hours per week	4	3.2	of which, course	2	3.3	applications	2
3.4	Total hours in the teaching plan	56	3.5	of which, course	28	3.6	applications	28

Individual study	Hours
Manual, lecture material and notes, bibliography	15
Supplementary study in the library, online and in the field	15
Preparation for seminars/laboratory works, homework, reports, portfolios, essays	8
Tutoring	5
Exams and tests	5
Other activities	0

3.7	Total hours of individual study	48
3.8	Total hours per semester	104
3.9	Number of credit points	4

4. Pre-requisites (where appropriate)

4.1	Curriculum	Physics, Mathematics, Informatics
4.2	Competence	Mathematic, Physics, Informatics, Measurement Techniques, Technical Drawing

5. Requirements (where appropriate)

5.1	For the course	Video projector, Blackboard, White and Colored Chalk
5.2	For the applications	Blackboard, Demonstrations on representative equipment from laboratory

6. Specific competences

Professional competences	**C1** - Operating with basic Mathematical, Engineering and Computer Science concepts (4 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
Cross competences	N/A

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

7.1	General objective	• To know the structure, operation and the bases of design of mobile mechanical systems that can be found in the structure of mechatronic systems and which integrates mechanical components, electrical, electronics and information technology. • To know the main types of mobile mechanical systems (mechanisms), the main problems related to their study, the terminology and the dedicated technical drawing language and also the specific aided design methods.
7.2	Specific objectives	• To communicate effectively in writing and orally with specialists from the field of mechanical engineering; • To use methods and systems for measuring functional parameters of various mobile mechanical systems; • To use mathematical concepts and the suitable methods and software packages to simulate various rigid and mobile mechanical systems. • To participate and apply the obtained knowledge in interdisciplinary research and design teams; • To analyze and interpret experimental data from the field of mechanical engineering; • To understand and to critically analyze technical solutions from the field of mechanical engineering.

8. Contents

8.1. Lecture (syllabus)		Teaching methods	Notes
1	Introduction. Hardware structure of mechatronic systems. Place and role of mechanisms and mechanical transmissions into their structure.	free exposure at blackboard combined with multimedia presentations	
2	Structural synthesis of mechanisms.		
3	Energy storage elements, elements for movement guidance, assembly elements.		
4	Kinematic analysis of the mechanisms.		
5	Analysis and synthesis of cam mechanisms.		
6	Mobile mechanical systems with profiled elements (levers and wheels): Malta Cross (Geneva) mechanism, ratchets, mechanisms with stellar elemets, locking mechanisms. Mechanisms oscillating and adjustable racing.		
7	Analysis and synthesis of gear mechanisms.		
8	Study the acting forces on the mechanisms from the structure of mechatronic systems. Determination of the inertia forces of the kinematic elements from the structure of these systems.		
9	The balancing of mechanisms and machines.		
10	Determination of the reaction forces in kinematic joints.		
11	Modeling the movement of mobile mechanical systems. The equations of motion of mobile mechanical systems. Applications: phases of operation of mobile mechanical systems; efficiency of mobile mechanical systems;		
12	Mechanisms and special transmissions in the field of precision engineering.		

	Logical and compliant mechanisms.		
13	Mechanisms for industrial robots.		
14	Adjusting the movement of the mechanisms and machines. Moderators. Uniformizing.		

Bibliography

1. Demian, Tr., Mecanisme de mecanica fina, EDP. Bucureşti, 1981.
2. Demian, Tr., s.a, Elemente constructive de mecanica fină, EDP, Bucureşti, 1984.
3. Handra-Luca, V., Mecanisme, Ed.UT Pres, Cluj-Napoca, 1981.
4. Olariu, V., s.a - Mecanică tehnică. Ed. Tehnică, Bucureşti, 1982.
5. Maties, V., s.a., Tehnologie şi educaţie mecatronică, Ed.Todesco, Cluj-Napoca, 2001.
6. Szekely, E., Dali, A., Mecanisme, Ed.UT Pres, Cluj-Napoca, 1993.
7. Tatar, O. s.a – Mini şi Microroboţi, Ed. Todesco, Cluj-Napoca, 2005.
8. Dudiţă, Fl., ş.a., Mecanisme articulate, inventică, cinematică, Ed.Tehnică, Bucureşti, 1989.
9. Voinea, R., ş.a., Introducere în mecanica solidului cu aplicaţii în inginerie, Ed.Academiei, Bucureşti, 1985.

8.2. Applications (Seminars)		Teaching methods	Notes
1	Identify the basic components from the structure of mobile mechanical systems (mechanisms). Develop the kinematic and constructive schemes;	free exposure at blackboard combined with multimedia presentations	
2	Elements of technical drawing. Modeling and simulation the functionality of mechanical structures; Constructive solutions of mobile mechanical systems. Type of materials used in mechanical components.		
3	Kinematic analysis - problems. Applications		
4	Cam mechanisms. Laws of motion. Operating conditions. Forces transmission.		
5	The study of complex gear trains. Determination of transmission ratios and angular velocities. Applications		
6	The balancing of mechanisms: four-bar mechanism and crank mechanism, the balancing of multi-cylinder engines. Applications		
7	The study of mechanisms and devices for uniform motion.		

Bibliography

1. Demian, Tr., s.a, Elemente constructive de mecanică fină, EDP, Bucureşti, 1984.
2. Handra-Luca, V., Mecanisme, Ed.UT Pres, Cluj-Napoca, 1981.
3. Handra-Luca, V., ş.a.– Introducere în teoria mecanismelor, Editura Dacia, Cluj-Napoca, vol. I-II, 1982, 1983.
4. Maros, D., ş.a. – Mecanisme, Indrumător de lucrări, Lito. I.P.C-N, Cluj-Napoca, 1984.
5. Olariu, V., s.a - Mecanică tehnică. Ed. Tehnică, Bucureşti, 1982.
6. Szekely, E., Dali, A., Mecanisme, Ed.UT Pres, Cluj-Napoca, 1993

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

The course curriculum exists in universities and faculties in the country and abroad. Its content is in conjunction with the expectations of community representatives, professional associations and employers in the field of Electronics and Telecommunication Engineering.
By learning theoretical concepts and addressing practical aspects included in the discipline entitled *Mechanical Elements and Mechanisms*, students acquire a consistent stock of knowledge, in accordance with partial competencies required for possible occupations provided in Grid 1 – RNCIS.

10. Evaluation

Activity type	10.1	Assessment criteria	10.2	Assessment methods	10.3	Weight in the final grade
Course		The course ends up with written and oral exam.		The mark is evaluated based on the score obtained at written exam and the answers given at oral examination.		60%
Applications		Every student gets a mark for his seminar activity.		The mark is evaluated based on the answers given at seminar activities		20%
		At the seminar meetings, students will receive individual homework. Through homework is required to		The homework must be presented by every student and a		

		analyze the structural and functional characteristics of a technical representative system and to identify the main modules for transmission and movement transformation.		mark is given accordingly.		20%
10.4 Minimum standard of performance						
The student must get at least a mark of 5 at every type of activity.						

Course responsible
Prof. dr. eng. Vistrian Mătieș

Head of department
Prof.dr.eng. Rodica Potolea