"ALEXANDRU IOAN CUZA"
UNIVERSITY OF IAŞI

FACULTY OF COMPUTER SCIENCE

Study Guide

2009/2010

I. GENERAL INFORMATION ABOUT THE FACULTY

1. Name and address

"Alexandru Ioan Cuza" University
Faculty of Computer Science
Str. General Berthelot nr. 16
700483 Iaşi
Romania
http://www.infoiasi.ro

2. Short history and mission

History

The interest for Computer Science at the "Alexandru Ioan Cuza" University of Iaşi dates back in 1958-1965, when Adolf Haimovici, professor at the Faculty of Mathematics, initiated a series of lectures about "Mathematics Applied in Computer Science"; in 1960, professor Haimovici taught the first course entitled "Elements of Computer Science".

In 1965, the Section of Computing Machines was set up within the Faculty of Mathematics; its first students graduated in 1970. In 1971, the section changed its name in the Section of Computer Science.

The year 1975 witnessed the creation of the Computing Centre of the University. During the leadership of professor Călin Petru Ignat, the first head of the Centre, were created the conditions for the development of research activity at the Computing Centre and thus for the emergence, in time, of the Faculty of Computer Science. Continuing this process of construction, professor Toader Jucan - head of the Centre since 1981 - pursued a rigorous policy regarding the presonnel's selection and training; many of today's professors of the faculty owe him their academic careers. The following heads of the Centre, especially Cornelius Croitoru and Gheorghe Grigoraş, continued on the same direction, until the Centre has been integrated in the Faculty of Computer Science.

As a result of the new realities - the significant growth of the number of students in computer science, the necessity for a continuous and quick adaptation of the teaching plans to the permanent changes in the field of computer science -, in the Autumn of 1991 the teaching staff of the Department of Computer Science of the Faculty of Mathematics proposed the creation of a new faculty, based on the structure of the existing section. In December, the Senate of the University aproved the creation of the new faculty. The representatives to the Senate of the future Faculty of Computer Science were, by that time, professor Călin Ignat (Rector of the University) and associate professor Cornelius Croitoru. Thorugh the decision of the Ministry of Education and Science, in January 1992 was set up the Faculty of Computer Science of the "Alexandru Ioan Cuza" University of Iaşi.

During the 17 years that have passed, the deans of the faculty have been, chronologically: prof. dr. Costică Cazacu (February-July 1992), assoc. prof. dr. Gheorghe Grigoraş (1992-1996), prof.dr. Călin Ignat (1996-1997), prof. dr. Toader Jucan (1997-2000), prof. dr. Dan Cristea (2000-2004), and prof. dr. Gheorghe Grigoraş (since 2004).

The initial structure of the faculty included two departments: the Department of Theoretical Computer Science and the Department of Applied Computer Science. Between 2004 and 2007 there were three departments: the Department of Computer Fundamentals and Distributed Systems, the Department of Optimization and Artificial Intelligence, and the Department of Software Systems. In 2007, as a consequence of a major reorganization of the University, the number of departments of the Faculty of Computer Science was reduced to one, called the Department of Computer Science.

Over 1000 students are currently studying at the Faculty of Computer Science.

Mission

The mission of the Faculty of Computer Science is to train highly effective specialists, capable of insuring the development of the Information Society in Romania, in the context of our country's integration within the structures of the European Community.

In order to achieve that mision, by promoting excellence in teaching and research activity, the Faculty of Computer Science does the following:

- Offers study programmes for the BSc diploma in Computer Science (both daily courses and distance learning), the diploma of Master in Computer Science, and the PhD diploma in Computer Science. The graduates of the faculty are capable of working as software developers, system engineers, network administrators, information system managers, etc., or as teachers of Computer Science.
- Provides, for each study programme, competitive study plans that allow the recognition of the diplomas in any other country.
- Coordinates the research activity in Computer Science, materialized through the publication of scientific papers in the Technical Report Series of the faculty, in the faculty's review (the Scientific Annals of the University - Computer Science series), and in national and international publications. Participation to scientific events inside and outside Romania, to research projects financed by the Romanian authorities and/or by the European Community, is also encouraged.
- Provides Computer Science courses for other faculties and within the Continuous Education system for the school and high school teachers.
- Participates in European programmes of student and teaching staff exchange.

The Faculty of Computer Science, together with the other faculties, through its entire activity, aims at the development of the local and regional community, at the growth of the "Alexandru Ioan Cuza" University's prestige, both at national and international level.

3. Administrative structure

Board of the Faculty

Dean: prof. dr. Gheorghe Grigoraş
Vice-deans:
- prof. dr. Dan Cristea
- assoc. prof. dr. Sabin Corneliu Buraga

Head of the Department of Computer Science: prof. dr. Dorel Lucanu
Chancellor: lect. dr. Vlad Rădulescu

Administrative staff

Chief administrator of the faculty: Radu Negrescu
Secretaries:
- Chief secretary: Maria Buburuzan
- Study situations: Lăcrămioara Leonte
- Fee problems: Mariana Nichita
- Distance learning: Lavinia Pîrîu
- Teaching problems: Alina Popescu

Phone: +4-0232-201090
Fax: +4-0232-201490
E-mail: `secret@infoiasi.ro`
Work with students - timetable: Monday to Thursday, 10-12

II. DEGREE PROGRAMMES

1. Qualifications awarded

In accordance to the Bologna system, the duration of BSc studies is 3 years. The graduates of the Faculty of Computer Science achieve the qualification of computer scientists.

The section of Distance Learning has been working since the academic year 2002-2003. The Faculty of Computer Science provides the frame for the continuous improvement of the school and high school teachers.

The MSc (Master) studies have a duration of 2 years. The oldest Master programme in the faculty is on Parallel and Distributed Computing, set up in 1995. Between 1995 and 1998, the section also had a French branch, where the courses were taught by French professors from the Universities of Paris Sud-Orsay, USTL Lille and Sorbonne.

The other Master sections currently functioning in the faculty are: Computational Linguistics (since 2001), Computational Optimization (since 2003), Software Engineering (since 2005), and Information Security (starting in 2009).

2. Admission requirements

Admission to the Faculty of Computer Science is open to all Romanian and EU citizens, as well as for non-EU people. The candidates are required to be high-school graduates and to have studied Mathematics and/or Computer Science in high school.

The admission process consists in the analysis of the personal records of the candidates. The final admission result is computed from the following components:

- the result of the Mathematics/Computer Science test within the high-school graduation exam (50%)
- the final result of the high-school graduation exam (20%)
- the mean of the study results during the high-school study years (30%)

3. Educational and professional goals

The main goal of the Faculty of Computer Science is to teach and train the students in order to become high-performance computer science professionals. This goal is achieved by providing strong theoretical foundations and, at the same time, extensive programming skills. Here are a few domains for which the students qualify after graduation:

- researcher in computer science
- software engineer
- programmer
- database architect/adminstrator
- system administrator

4. Examination and assessment regulations

The Romanian system comprises marks from 1 to 10. The lowest mark required for passing an exam is 5.

There is a major interest in supporting the continuous evaluation of the students, along with the teaching process. According to the regulations of the "Alexandru Ioan Cuza" University, at least 50% of the final result on each discipline must consist in the assessment of the activity carried out by students during the semester. That is, there are normally two parts of the examination:

- The continuous assessment, during the semester. This may be implemented either by projects that students must fulfill before certain deadlines are reached, or by written/practical tests.
- The final exam, which is sustained within the examination session (at the end of the semester), usually as a written test. The teacher may decide that continuous assessment is sufficient, in which case there is no final exam.

After all parts of the examination are sustained, an overall score is computed for each student. The final result is determined as follows:

- For the scores below the promotion threshold, the final result is between 1 and 4 (failed).
- For the scores above the promotion threshold, the final result is between 5 and 10 and is computed by applying the Gauss distribution over all scores.

5. Final examination

For BSc students, the final examination comprises two components:
- Fundamental knowledge of computer science
- Defending the diploma thesis

The final result is the average of the marks obtained for the two components.

For Master students, the final examination consists in defending the dissertation thesis.

For both BSc and Master students, the evaluation is made by an examination comission. The thesis supervisor is a rightful member of the comission.

6. Access to further studies

BSc graduates may follow a Master specialization, after going through an admission contest. The Master modules of the Faculty of Computer Science are also open for the graduates of other faculties; at the same time, graduates of the Faculty of Computer Science may follow Master modules of other faculties.

Master graduates may continue with PhD studies. The doctoral programme has been included in the structure of the faculty since 1993. There are currently 7 PhD supervisors in the Faculty of Computer Science, each one with his/her own fields of interest.

7. Coordination of teaching activity

ECTS coordinator: assoc. prof. dr. Sabin Corneliu Buraga
Erasmus coordinator: lect. dr. Vlad Rădulescu

8. Study plans

Undergraduate

No	Code	Name	Hours per week				Credits	Evaluation			
			C	S	L	Pr.		P	C	E	Mixed
Semester I (Year I)											
1	CS1101	Algorithms and Programming	2	1	1		5				X
2	CS1102	Computer Architecture and Operating Systems	2	1	1		5				X
3	CS1103	Logics for Computer Science	2	2			5				X
4	CS1104	Mathematics	2	2			5				X
5	CS1105	Communication in Electronic Environments	2	1			5				X
6	CS1106	English Language I		2			5				X
Semester II (Year I)											
7	CS1207	Object-Oriented Programming	2		2		5				X
8	CS1208	Operating Systems	2		2		5				X
9	CS1209	Algebraic Foundations of Computer Science	2	2			5				X
10	CS1210	Probabilities and Statistics	2	1	1		5				X
11	CS1211	Hardware Practice	1		3		5				X
12	CS1212	English Language II		2			5				X
Semester III (Year II)											
13	CS2101	Computer Networks	2		2		5				X
14	CS2102	Databases	2		2		5				X

No	Code	Name	Hours per week				Credits	Evaluation			
			C	S	L	Pr.		P	C	E	Mixed
15	CS2103	Formal Languages, Automata and Compilers	2	2			5				X
16	CS2104	Graph Algorithms	2	2			5				X
17	CS2105	Optional package 1	2		2		5				X
18	CS2106	English Language III		2			3				X
Optional Disciplines											
	CS2105O1	Coding Theory and Cryptography	2		2		5				X
	CS2105O2	Game Theory	2		2		5				X
	CS2105O3	Continuous Models and Matlab	2		2		5				X
Semester IV (Year II)											
19	CS2207	Web Technologies	2		2		5				X
20	CS2208	Advanced Programming Techniques	2		2		5				X
21	CS2209	Software Engineering	2		2		5				X
22	CS2210	DBMS Practice	1		2		5				X
23	CS2211	Optional package 2	2	2			5				X
24	CS2212	English Language IV		2			3				X
Optional Disciplines											
	CS2211O1	Logic Programming	2	2			5				X
	CS2211O2	Functional Programming	2	2			5				X
Semester V (Year III)											
25	CS3101	Design and Analysis of Algorithms	2	2	0		5				X
26	CS3102	Information Security	2	1	1		5				X
27	CS3103	Artificial Intelligence	2	0	2		5				X
28	CS3104	Application Development on .NET Platform	2	0	2		5				X
29	CS3105	Optional package 3	2		2		5				X
30	CS3106	Practice	1		3		5				X
Optional Disciplines											
	CS3105O1	Stochastic Processes	2		2		5				X
	CS3105O2	Computability, Decidability and Complexity	2		2		5				X
Semester VI (Year III)											
31	CS3207	Numerical Calculus	2		2		5				X
32	CS3208	Computer Graphics	2		2		5				X
33	CS3209	Optional package 4	2		2		5				X
34	CS3210	Optional package 5	2		2		5				X
35	CS3211	Optional package 6	2		2		5				X
36	CS3212	Diploma Paper Work			4		5				X
Optional Disciplines											
	CS3209O1	Rule Based Programming	2		2		5				X
	CS3209O2	Bioinformatics	2		2		5				X
	CS3210O1	The Modelling of Distributed Systems Using Petri Nets	2		2		5				X
	CS3210O2	Genetic Algorithms	2		2		5				X
	CS3211O1	Advanced Operating Systems	2		2		5				X

No	Code	Name	Hours per week				Credits	Evaluation			
			C	S	L	Pr.		P	C	E	Mixed
	CS3211O2	Embedded Systems	2		2		5				X

Master in Distributed Systems

No	Code	Name	Hours per week				Credits	Evaluation			
			C	S	L	Pr.		P	C	E	Mixed
Semester I (Year I)											
1	MSD1101	Advanced Software Engineering Technics	2		2		8				X
2	MSD1102	Operational Research	2		2		8				X
3	MSD1103	Java Technologies	2		2		8				X
4	MSD1104	Research Project I			4		6				X
Semester II (Year I)											
5	MSD1205	Advanced Artificial Intelligence	2		2		8				X
6	MSD1206	Parallel Algorithms and Parallel Programming	2		2		8				X
7	MSD1207	Algorithms and Distributed Programming	2		2		8				X
8	MSD1208	Research Project II			4		6				X
Semester III (Year II)											
9	MSD2101	Web Applications Development	2		2		8				X
10	MSD2102	Distributed Systems Modeling	2		2		8				X
11	MSD2103	Distributed Operating Systems	2		2		8				X
12	MSD2104	Research Project III			4		6				X
Semester IV (Year II)											
13	MSD2206	Project Management	2		2		8				X
14	MSD2205	Specification and Verification of Concurrent and Distributed Systems	2		2		8				X
15	MSD2207	Security Protocols	2		2		8				X
16	MSD2208	Dissertation Preparing			4		6				X

Master in Computational Optimization

No	Code	Name	Hours per week				Credits	Evaluation			
			C	S	L	Pr.		P	C	E	Mixed
Semester I (Year I)											
1	MOC1101	Advanced Software Engineering Technics	2		2		8				X
2	MOC1102	Operational Research	2		2		8				X
3	MOC1103	Java Technologies	2		2		8				X
4	MOC1104	Research Project I			4		6				X
Semester II (Year I)											
5	MOC1205	Advanced Artificial Intelligence	2		2		8				X
6	MOC1206	Combinatorial Optimization	2		2		8				X
7	MOC1207	Nature Inspired Methods	2		2		8				X
8	MOC1208	Research Project II			4		6				X
Semester III (Year II)											
9	MOC2101	Web Applications Development	2		2		8				X

No	Code	Name	Hours per week				Credits	Evaluation			
			C	S	L	Pr.		P	C	E	Mixed
10	MOC2102	Special Chapters in Artificial Intelligence	2		2		8				X
11	MOC2103	Experimental Analysis of Algorithms	2		2		8				X
12	MOC2104	Research Project III			4		6				X
Semester IV (Year II)											
13	MOC2205	Project Management	2		2		8				X
14	MOC2206	Neural Networks	2		2		8				X
15	MOC2207	Data Mining	2		2		8				X
16	MOC2208	Dissertation Preparing			4		6				X

Master in Software Engineering

No	Code	Name	Hours per week				Credits	Evaluation			
			C	S	L	Pr.		P	C	E	Mixed
Semester I (Year I)											
1	MISS1101	Advanced Software Engineering Technics	2		2		8				X
2	MISS1102	Operational Research	2		2		8				X
3	MISS1103	Java Technologies	2		2		8				X
4	MISS1104	Research Project I			4		6				X
Semester II (Year I)											
5	MISS1205	Advanced Artificial Intelligence	2		2		8				X
6	MISS1206	Multimedia Technologies	2		2		8				X
7	MISS1207	Special Chapters in Human-Computer Interraction	2		2		8				X
8	MISS1208	Research Project II			4		6				X
Semester III (Year II)											
9	MISS2101	Web Applications Development	2		2		8				X
10	MISS2102	Formal Methods in Software Engineering	2		2		8				X
11	MISS2103	Software Security	2		2		8				X
12	MISS2104	Research Project III			4		6				X
Semester IV (Year II)											
13	MISS2205	Project Management	2		2		8				X
14	MISS2206	Advanced Databases	2		2		8				X
15	MISS2207	Software Quality	2		2		8				X
16	MISS2208	Work on the Dissertation Thesis			4		6				X

Master in Computational Linguistics

No	Code	Name	Hours per week				Credits	Evaluation			
			C	S	L	Pr.		P	C	E	Mixed
Semester I (Year I)											
1	ML1101	Advanced Software Engineering Technics	2		2		8				X
2	ML1102	Operational Research	2		2		8				X
3	ML1103	Java Technologies	2		2		8				X
4	ML1104	Research Project I			4		6				X

No	Code	Name	Hours per week				Credits	Evaluation			
			C	S	L	Pr.		P	C	E	Mixed
Semester II (Year I)											
5	ML1205	Advanced Artificial Intelligence	2		2		8				X
6	ML1206	Introducing Natural Language Processing	2		2		8				X
7	ML1207	Natural Language Processing by Statistical Methods	2		2		8				X
8	ML1208	Research Project II			4		6				X
Semester III (Year II)											
9	ML2101	Web Applications Development	2		2		8				X
10	ML2102	Special Chapters in Artificial Intelligence	2		2		8				X
11	ML2103	Ontologies in Natural Language Processing	2		2		8				X
12	ML2104	Research Project III			4		6				X
Semester IV (Year II)											
13	ML2205	Project Management	2		2		8				X
14	ML2206	Speech Processing Tehnologies and Fuzzy Systems	2		2		8				X
15	ML2207	Multilingual Web and Machine Translation	2		2		8				X
16	ML2208	Dissertation Practice			4		6				X

Master in Information Security

No	Code	Name	Hours per week				Credits	Evaluation			
			C	S	L	Pr.		P	C	E	Mixed
Semester I (Year I)											
1	MSI1101	Advanced Software Engineering Technics	2		2		8				X
2	MSI1102	Operational Research	2		2		8				X
3	MSI1103	Java Technologies	2		2		8				X
4	MSI1104	Applied Cryptography			4		6				X
Semester II (Year I)											
5	MSI1205	Models of Security	2		2		8				X
6	MSI1206	Advanced Artificial Intelligence	2		2		6				X
7	MSI1207	Network Security	2		2		8				X
8	MSI1208	Security Protocols	2		2		8				X
Semester III (Year II)											
9	MSI2101	Software Security	2		2		8				X
10	MSI2102	Wireless and Mobile Security	2		2		8				X
11	MSI2103	Web Applications Development	2		2		8				X
12	MSI2104	Security of Operating Systems			2		3				X
12'	MSI2104'	Malicious Software			2		3				
Semester IV (Year II)											
13	MSI2205	Belief Logics in Information Security	2		2		8				X
14	MSI2206	Security of Electronic Commerce	2		2		8				X
15	MSI2207	Project Management	2		2		8				X
16	MSI2208	Dissertation Preparing			4		6				X

Hours per week:
C - course
S - seminary
L - laboratory
Pr - project

Evaluation:
P - during the semester
C - oral examination
E - written examination

9. Course descriptions

As the Master modules have a series of common courses, only one course description was provided for each such common discipline. Unique course codes are used for these courses:

- MCG1101 for MSD1101, MOC1101, MISS1101, ML1101, and MSI1101
- MCG1102 for MSD1102, MOC1102, MISS1102, ML1102, and MSI1102
- MCG1103 for MSD1103, MOC1103, MISS1103, ML1103, and MSI1103
- MCG1205 for MSD1205, MOC1205, MISS1205, ML1205, and MSI1206
- MCG2101 for MSD2101, MOC2101, MISS2101, ML2101, and MSI2103
- MCG2205 for MSD2205, MOC2205, MISS2205, ML2205, and MSI2207

COURSE NAME	**ALGORITHMS AND PROGRAMMING**	CODE: CS1101

STUDY YEAR	I	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	1	1	-	56	94	5	M	Romanian

COURSE TEACHERS	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	LECT. DR. CRISTIAN GAȚU LECT. DRD. MARIAN BALTĂ	Computer Science

PREVIOUS COURSES REQUESTED	-

OBJECTIVES	Algorithms: Techniques for the design and analysis of efficient algorithms, introduction to a number of fundamental data structures and algorithms (or procedures) for manipulating them, a basic understanding of how common computational problems can be solved efficiently on a computer. Programming: basic skills in using an imperative programming language (C), basic concepts and techniques of program design, the evaluation of the run time.
GENERAL DESCRIPTION	Algorithms: algorithmic language, static data structures, dynamic data structures, linear lists, trees, graphs (as data structures), heaps, union-find, sorting, searching, problem solving. Programming: Gradual presentation of the C language (ISO Standard), focusing on the efficient implementation of the fundamental data structures and the case studies presented in the first part.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Seminar: problem solving using basic algorithms and fundamental data structures Laboratory: Fundamental data types and control flow. Files and input-output operations. Arrays and pointers. Functions and macros. Implementation of the fundamental data structures.
TEACHING METHODS	Transparences and video projector.

BIBLIOGRAPHY (SELECTION)	D. Lucanu: Bazele proiectării programelor si algoritmilor, Universitatea "Al. I. Cuza", Iasi, 1996 D. Lucanu: Proiectarea algoritmilor - Tehnici elementare, Editura Universității "Al. I. Cuza", Iași, 1993 T.H. Cormen, C.E. Leiserson, R.L. Rivest: Introducere in Algoritmi, Computer Libris Agora, 2000 Al Kelley, Ira Pohl: A Book on C - Programming in C, Addison Wesley, Reading Herbert Schildt: C Manual Complet, Bucuresti, Ed. Teora 1998 E. Horowitz, S. Sahni, S. Anderson - Freed: Fundamentals of Data Structures in C, Computer Science Press, 1993

EVALUATION		
	conditions	Seminary Activity(AS), Laboratory Activity (AL), Written Tests (TS)
	criteria	AS $\geq$ 5, AL $\geq$ 6, TS $\geq$ 4
	evaluation methods	AS: questions, participation at discussions, original solutions to problems. AL: each topic is evaluated with a mark between 1 and 10. TS: 2 written tests (weeks 7, 13), each test including 8 questions and a problem.
	final result - formula	Final Mark = 10% AS + 40% AL +50% TS

COURSE NAME	**COMPUTER ARCHITECTURE AND OPERATING SYSTEMS**	CODE: CS1102

STUDY YEAR	I	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	1	1	-	56	94	5	M	Romanian

COURSE TEACHERS	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. HENRI LUCHIAN LECT. DR. VLAD RĂDULESCU	Computer Science

PREVIOUS COURSES REQUESTED	-

OBJECTIVES	At the end of the semester, the students should know: - the basic elements of the computer architecture and organization - the fundamentals of internal data representation in computers - the main functions of an operating system
GENERAL DESCRIPTION	Von Neumann architecture. Empirical laws. Memory hierarchies. Physical representation of information; bits and bytes. Boolean functions; minimization. Logic gates and combinational circuits; adder, decoder. Flip-flops and sequential circuits; counter, shift register. Internal data representation. Fixed point representations. Floating point representations. The memory; technology, cache memory, virtual memory. The structure of the central processing unit. Techniques for performance improvement. Pipeline. RISC architecture. Parallel architectures. Peripheral devices. The interrupt system. Introduction to the operating systems. Kernel and drivers. System calls. Process management. Memory management; physical and virtual addresses, segmentation and pagination. Creating and executing programs.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Empirical laws. Amdahl's law. Locality laws. Boolean functions. Minimization techniques. Fixed point representations. Overflow. Floating point representations. Overflow. Assembly language of the x86 microprocessor family.
TEACHING METHODS	Exposition, debate, problem-solving, case studies, exercises.

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

EVALUATION	conditions	The presence at the laboratoty and seminary activities.
	criteria	At least 5 points on each test; 3 or 4 points are accepted for at most one test.
	evaluation methods	Two written tests: the first (TS1) during the 7th week (40 hours individual study), the second (TS2) during the examination session (40 hours individual study). One practical test (TL) during the 14th week, regarding the laboratory activity (14 hours individual study).
	final result - formula	If at least 5 points were achieved on each of the three tests: Final result = (TS1 + TS2 + TL) / 3 If 4 points were achieved on one of the tests: Final result = (TS1 + TS2 + TL) / 3, with truncation If 3 points were achieved on one of the tests: Final result = (TS1 + TS2 + TL) / 3 - 1, with truncation

COURSE NAME	**LOGICS FOR COMPUTER SCIENCE**	CODE: CS1103

STUDY YEAR	I	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	2	-	-	56	94	5	**M**	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. CRISTIAN-DUMITRU MASALAGIU	Computer Science

PREVIOUS COURSES REQUESTED	-

OBJECTIVES	1. Getting minimal skills for using Logic in Computer Science and understanding its importance 2. Understanding the needed fundamental concepts for the future study of related disciplines such as: Logic Programming, Specification and Verification of Real Systems, Expert Systems Web Ontologies or Non-classical Logics 3. Getting minimal skills to rigorously work with software applications and media
GENERAL DESCRIPTION	1. Logic in Computer Science (introduction, motivation) 2. Boolean Algebras (semantic domains) 3. Propositional Logic (**LP**) 4. First-order Predicate Calculus (**LP1**) 5. Introduction to Deductive Systems and Logical Theories 6. Introduction to Logic Programming 7. The Ideea of Verification
DESCRIPTION OF SEMINARY / LABORATORY WORKS	The main goal of any Seminar is to facilitate a deeper understanding of the content of the prevoius Courses, with the help of more complicated, new and detailed examples. This will be accomplished with the direct participation of the students
TEACHING METHODS	All the classical didactic methods will be used: systematic exposure of knoledge, conversation, learning „by descovery", etc. The Courses will be taught using a video-projector

MAIN BIBLIOGRAPHY (SELECTION)	
	1. C.D. Masalagiu – Fundamentele logice ale Informaticii, Editura Universității "Al. I. Cuza", Iași, 2004, ISBN 973-703-015-X (in Romanian) 2. C. Cazacu, V. Slabu – Logică Matematică, Editura "Ştefan Lupaşcu", Iași, 1999, ISBN 973-99044-0-8 (in Romanian) 3. Specific INTERNET sites

EVALUATION		
	conditions	Every student will be tested 1 to 4 times during the 14-th seminaries (see bellow), not necessarily immediatelly to the corresponding course. A bonus may be granted for supplementary (good) answers
	criteria	The above examinations can produce a maximum of 60 points. The exam (in the special session of the 7^{th} and 14^{th} week) may produce other (maximum) 60 points. To "graduate" the course, a minimum of 40 points is need
	evaluation methods	Additional written test may be given at fixed or no-announced dates. Thus, the total amount of points needed to pass is between 40 and 120.
	final result - formula	The final grade is computed by first summing up all the obtained poins and then by dividing the result by ten (between 40 to 50 points the degree will be 5.00). The grades will be then rounded such as to get a Gauss curve for the given year of study (see the regulations). The grades grater than 10 will be rounded to ten.

COURSE NAME	**MATHEMATICS**	CODE: CS1104

STUDY YEAR	I	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	2	-	-	56	94	5	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	LECT. DR. FLORIN IACOB	Computer Science

PREVIOUS COURSES REQUESTED	Mathematical Analysis, Analythical Geometry and Algebra (from high-school)

OBJECTIVES	• To systematize and to get thoroughly into the study of some theoretical and applied questions on differential and integral calculus in the spaces R, $\overline{R}$ and R^n ($n \geq 2$).To present certain basic results and applications in the domain of real, differentiable and integrable, scalar and vectorial functions. • To point out fundamental ideas in real analysis, algebra and geometry of curves and surfaces for the benefit of other objects of study at the Faculty of Computer Science.
GENERAL DESCRIPTION	Elements of algebra, analysis (topology) and geometry in connection with the spaces R, $\overline{R}$ and R^n ($n \geq 2$). Numerical sequences and series. Functions of one and several real variables (generalities, limits, continuity, derivatives, differentials, series and integrals). Basic applications.
DESCRIPTION OF SEMINARY WORKS	Sets, relations and functions (generalities). Basic algebraical structures. Numerical real sets. Remarkable numerical inequalities. Sequences and series of real numbers. Algebraic and topological aspects of the space R^n ($n \in N^*$). Real functions (generalities). Linear, affine and quadratic real forms (algebraic considerations and geometrical interpretations) . Limits and continuity for real functions. Derivatives, differentials and its applications (implicit functions, local inverse functions, functional dependence or independence, unconditioned and conditioned extreme problems). Sequences and series of real functions (series of powers, Taylor and Fourie series). Simple and multiple, definite and indefinite, without or with parameters Riemann integrals.
TEACHING METHODS	Verbal and written exposure of the most important definitions, results and applications. Heuristic approach using the course and seminary resources – supplied on a special website (http://thor.info.uaic.ro/~fliacob/An1/2007-2008).

BIBLIOGRAPHY (SELECTION)	• Protter H. Murray – *Basic elements of real analysis*, 1998. • Precupanu Anca – *The Basis of Mathematical Analysis* (in Romanian), Ed. Polirom, Iaşi,1998. • Rodica Luca-Tudorache – *Mathematical Analysis. Differential Calculus* (in Romanian), Ed. Tehnopress, Iaşi, 2005. • F. Iacob – *Mathematics* (Romanian Course and Seminary Notes; on the mentioned site), 2008. • D. Buşneag, Dana Piciu – *Algebra Lectures* (in Romanian), Ed. Universitaria, Craiova, 2002. • Narcisa Apreutesei-Dumitriu, Gabriela Apreutesei – *Introduction in the Theory of Integrability* (in Romanian), Ed. Performantica, Iaşi, 2005. • V. Postolică – *Efficiency by Mathematical Analysis* (Rom.), Ed. Matrix Rom, Bucureşti, 2006.

EVALUATION	conditions	Compulsory participation at the written examinations during the semester and the exam.
	criteria	Presence and activity at the seminaries. Home works achievement. Written examination results. Participation in councelling programme.
	evaluation methods	Appreciation by marks on the frequency (NFS1,NFS2) and activity at the seminaries (NPS1,NPS2), homeworks (NT1,NT2), written examinations (NL1,NL2) and participation in councelling programme (NPC1, NPC2).
	final result–formula and the pass mark	PF = (0,75*NFS1+NPS1+ 1,25*NET1+1,75*NL1+0,25*NPC1)+ (0,75*NFS2+ NPS2+ 1,25*NET2+1,75*NL2+0,25*NPC2) $\geq$ 45; The final mark (NF) is established from PF, in according with the new ECTS norms.

COURSE NAME	**COMMUNICATION IN ELECTRONIC ENVIRONMENTS**	CODE: CS1105

STUDY YEAR	I	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	1	-	-	42	108	5	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	LECT. DR. MIHAELA BRUT	Computer Science

PREVIOUS COURSES REQUESTED	-

OBJECTIVES	Development of the abilities for structuring a spoken / written discourse, in different communication contexts and situations. How to organize and present the ideas, how to format the documents. Understanding of the Internet functioning mode, and usage of its main services. Web sites development, by using HTML and CSS according Web design principles.
GENERAL DESCRIPTION	• Internet on+line environments. Communication through on-line means. World Wide Web. • XHTML - eXtensible HyperText Markup Language • CSS - Cascading Style Sheets • Web resources search and retrieval • Web sites design and development • Social Web (Web 2.0) • Electronic documents. Microformats • On-line communities
DESCRIPTION OF SEMINARY / LABORATORY WORKS	• Thematic Web sites design, and case studies for different Web sites types: presentation Web sites, e-commerce, e-learning, web portals • Tools for Web sites design and development • Using various social Web applications. Integrating in different on+line communities • MS Office: using and defining of various documents types templates. Document formatting, text processing • Organizing, conceiving, and presentation of different types of discourses, papers, writings, documents: CV, tutorial, degree thesis, book/product/company presentation
TEACHING METHODS	Tutorial exposition, case studies, dialogue.

BIBLIOGRAPHY (SELECTION)	1. J. Beaird, The Principles of Beautiful Web Design, Sitepoint, 2007 2. C. Bertrand (coord.), O introducere în presa, Polirom, 2001 3. M. Brut, Instrumente pentru E-Learning, Polirom, 2006 4. S. Buraga, Proiectarea siturilor Web (editia a II-a), Polirom, 2005 5. K. Carey, S. Blatnik, Design Concepts with Code: A Developer Approach, Apress, 2003 6. L. Goin, Design for Web Developers: Colour and Layout for the Artistically Overwhelemed, DMXzone.com, 2005 7. P. Haine, HTML Mastery: Semantics, Standards, and Styling, Friends Of ED, 2006 8. R. Hoff, Regulile unei prezentari de succes, Curtea Veche, 2002 9. A. de Peretti, J.-A. Legrand, J. Boniface, Tehnici de comunicare, Ed. Polirom, Iaşi, 2001 10. S. Prutianu,Antrenamentul abilităţilor de comunicare, Ed. Polirom, Iaşi, 2004 11. L. Wroblewski, Site-Seeing – A Visual Approach to Web Usability, Hungry Minds, 2002.

EVALUATION		
	conditions	Minimum 50 points for each of the two projects
	criteria	Spoken / written discourse structuring, complying with the design principles, originality
	evaluation methods	Project, activity in practical works, systematic involving observation
	final result - formula	PF = 0.4*P1 + 0.4*P2 + 0.1*AL + 10, where PF= final score, P1=score for project 1, P2=score for project 2, AL=score for practical works activity. The final mark will be established according the Gauss diagram.

COURSE NAME	**ENGLISH**	CODE: CS1106

STUDY YEAR	I	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
0	2	0		28	122	5	M	ENGLISH

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	DRD. DRAGOS ZETU	Computer Science

PREVIOUS COURSES REQUESTED	

OBJECTIVES	To improve the student's proficiency in English in general, computer science English in particular.
GENERAL DESCRIPTION	The seminars will provide the student the possibility to work with authentic materials, interact and improve his/her grammar skills.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	The seminars will provide the student the possibility to work with authentic materials, interact and improve his/her grammar skills.
TEACHING METHODS	

BIBLIOGRAPHY (SELECTION)	Any English grammar compendium

EVALUATION		
	conditions	Attendance, written exam
	criteria	Good proficiency in english
	evaluation methods	Continuous evaluation, written exam
	final result - formula	50% exam results, 50% seminar activity

COURSE NAME	**OBJECT ORIENTED PROGRAMMING**	CODE: CS1207

STUDY YEAR	III	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	94	5	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. DOREL LUCANU PROF. DR. GHEORGHE GRIGORAS	Computer Science

PREVIOUS COURSES REQUESTED	Algorithms and Programming (CS1101)

OBJECTIVES	Teaching fundamental concepts in object-oriented programming and the use of the programming language C + +.
GENERAL DESCRIPTION	Language C++ (ISO Standard) with emphasis on the representation of objects and classes in C++, the relationship of derivation and description of hierarchies of classes in C++, virtual functions and implementation of polymorphism in C++, templates and parametric classes, using standard library. Concepts and principles in Object Oriented Programming: classes, objects, hierarchies of class, polymorphism, abstract classes, interfaces, parametric classes, observer pattern, composite pattern, iterator pattern, case studies.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Introduction classes in C + +, shifting from C to C + +, advanced classes, inheritance, polymorphism, templates. parametric classes, STL: Standard Template Library.
TEACHING METHODS	Slides with course items; seminar themes; projects' issues; electronic version of the course; main readings will be find on the web page

BIBLIOGRAPHY (SELECTION)	
	Herbert Schildt: C++ Manual Complet, Bucuresti, Ed. Teora 2000 D. Kaler, M.J. Tobler, J. Valter: C++, Teora, 2000 Bjarne Stroustrup: The C++ Programming Language, Adisson-Wesley, 3nd edition, 1997 Stanley B. Lippman: C++ Primer, Addison Wesley, 1992

EVALUATION		
	Conditions	Participation to laboratory hours (LA), participation to writing tests (WT)
	Criteria	LA >= 6, WT >= 4
	Evaluation methods	Mixed (during the semester and examination)
	Final result – formula	40% LA + 60% WT

COURSE NAME	**OPERATING SYSTEMS**	CODE: CS1208

STUDY YEAR	1	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS OF INDIVIDUAL ACTIVITY	CREDITS	EVALUATION TYPE (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	0	2	0	56	94	5	P	Romanian

TAUGHT BY	ACADEMIC AND SCIENTIFIC TITLE, NAME	DEPARTMENT
	LECT. DR. CRSTIAN VIDRAŞCU	Computer Science

REQUIRED COURSES	Computer Architecture and Operating Systems, C Programming

OBJECTIVES	The students who will attend this course will obtain knowledge about operating systems, regarding the techniques used for the design and the implementation of them. Also, they will obtain the skills to write parallel processing programs and to use the UNIX/Linux operating system.
GENERAL THEMATICS	Basic concepts about operating systems. Architecture, components, services. System kernel. Process management. Concurrency. Scheduling. Parallel processing. Process coordination. Interprocess communication. Main memory management. Memory hierarchy. Allocation methods. Segmentation and paging. Virtual memory. Cache memory. Secondary-storage management. File systems. Disk management. Distributed systems. Distributed coordination. Types of distributed operating systems. Distributed file systems.
SEMINARY / LABORATORY THEMATICS	Overview of the UNIX/Linux operating system. Structure and general features. Guide of usage. Commands. Shells. Bash scripting language. Concurrent programming in C language under Linux. Working with files. Exclusive/concurrent access to files. Process management. Creation, synchronization and executable loading. UNIX signals. Interprocess communication. Pipes. Fifos. Other communication mechanisms. Terminal management. Ncurses library.
TEACHING METHODS	Exposure using video-projector, demos on blackboard and on computer.

BIBLIOGRAPHY	1. C.Vidraşcu : Sisteme de operare. Manual pentru ID, Edit. Univ. „Al. I. Cuza", Iaşi, 2004. 2. C.Moroşanu, S.Pavăl : Sisteme de operare. Instalare, programare, utilizare LINUX. Edit. Univ. „Al. I. Cuza", Iaşi, 2006. 3. F.M.Boian et al. : Sisteme de operare. Edit. Risoprint, Cluj-Napoca, 2006. 4. A.Tanenbaum : Modern Operating Systems (Third Edition), Prentice-Hall, 2001. 5. A.Silberschatz et al. : Operating System Concepts (Sixth Edition), Addison-Wesley, 2001. 6. M.Ben-Ari : Principles of Concurrent Programming, Prentice-Hall International, 1982. 7. R.Stevens : Advanced UNIX Programming in the UNIX Environments, Addison-Wesley, 1992.

EVALUATION	conditions	The presence at the laboratory activities and at the written theses.
	criterias	Minimal score for graduation: TS1+TS2 $\geq$ 25p , L $\geq$ 10p
	modes	Evaluation during the semester: lab works and two written theses.
	formula	Final Score = Lab * 0.4 + WrittenThesis1 * 0.3 + WrittenThesis2 * 0.3. The final mark is obtained from the final score through classification based on the ECTS – European Credit Transfer System and Diploma Supplement.

COURSE NAME	**ALGEBRAIC FOUNDATIONS OF COMPUTER SCIENCE**	CODE: CS1209

STUDY YEAR	I	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	2	-	-	56	94	5	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. FERUCIO LAURENȚIU ȚIPLEA	Computer Science

PREVIOUS COURSES REQUESTED	No prerequisite required.

OBJECTIVES	The course deals with those topics from mathematics that have proven to be particularly relevant to students in computer science.
GENERAL DESCRIPTION	The course covers basic elements on set theory (sets, relations and functions, induction and recursion), partially ordered sets (posets, lattices), number theory (divisibility, prime numbers, congruences), semigroups and monoids, groups, rings, vector spaces and linear mappings. Relevant applications of each chapter of the course to computer science, are disccused.
DESCRIPTION OF SEMINARY / LABORATORY WORK	Seminars and laborators are grouped around the chapter currently discused in the course. They are aimed to illustrate the topics of the chapter mainly by practical applications.
TEACHING METHODS	On-line and blackboard presentation.

BIBLIOGRAPHY (SELECTION)	• H. Cohen. *A Course in Computational Algebraic Number Theory*, Springer-Verlag, third printing, 1996. • P.M. Cohn. *Algebra*, vol. 1, John Wiley & Sons, 1982. • P.M. Cohn. *Algebra*, vol. 2, John Wiley & Sons, 1989. • P.M. Cohn. *Algebra*, vol. 3, John Wiley & Sons, 1991. • F.L. Tiplea. *Introduction to Set Theory*, "Al.I.Cuza" University Press, 1998.

EVALUATION		
	conditions	To pass the course, at least 50 points out of 100 are needed from both the evaluation during the semester and from the final exam.
	criteria	
	evaluation methods	Evaluation during the semester and final exam.
	final result - formula	50% evaluation during the semester and 50% final exam.

COURSE NAME	**PROBABILITIES AND STATISTICS**	CODE: CS1210

STUDY YEAR	I	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	1	1	-	56	94	5	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	ASSOC. PROF. DR. SILVIA LUCHIAN	Computer Science

PREVIOUS COURSES REQUESTED	Communication in Electronic Environments (MS Office).

OBJECTIVES	Students should be able to apply fundamental probabilistic models andmethods to solve problems related to the study of random phenomena; they should also be able to use the computer in order to apply statistical methods for decision making, based on experimental data. Students should be able to recognise various types of random variables and to understand which variable can be used in given situations; they should also understand the basics of statisical reasoning, specifically to select and apply appropriate statistical tests for hypothesis testing.
GENERAL DESCRIPTION	**Descriptive Statistics**: synthesis and presentation of experimental data (see seminar below). **Probability Theory**: Events, operations over events. Conditional probability. The formula of Bayes. Random variables – repartition, operations, taxonomy. Discrete variables (bynomial, Poisson, geometric hipergeometric), continues variables (uniform, normal, exponential, gamma, Student, Chi Square,Weibull ,f). Moment generating functions. Vectors of random variables. Covariance. Corelation coefficient. Markov and Cebâşev inequalities. The strong law of large numbers. The Central limit theorem. Stochastic processes. **Inferential Statistics**: Parameter estimation. Confidence intervals for poplation parameters. Hypothesis testing for means, proportions, dispersions. Inferences on multinomial experiments. Inferences over two populations. Dispersional analysis.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Students will practice, using EXCEL®, how to solve specific real world problems using notions from Probability Theory, as well as various methods of Descriptive Statistics for organising and presenting raw data (relative and cumulative frequencies; proportions; frequency distributions; graphical representation of random variables; measures of central tendency; measures of variation), as well as raw data processing for statistical analysis : confidence intervals for means, propotions, dispersions; significance tests for means, proportions, disperions, including non-normal populations; inferences for two populations; qualitative variables; the Chi-square test (independence, homogenity); dispersional analysis.
TEACHING METHODS	exposition, problem-solving, case studies, exercise.

BIBLIOGRAPHY (SELECTION)	Johnson, R. : Elementary Statistics, PWS Publishers - Duxbury Press, Boston, 1991 (available in the Mathematics Library) Ciucu, Gh., Craiu, V.: Introducere în Teoria probabilităţilor şi Statistică matematică, Editura Didactică şi Pedagogică. Ciucu, Gh., Craiu, V., Săcuiu, I.: Probleme de Teoria probabilităţilor, Editura Tehnică. Blattner, P.: Microsoft®EXCEL, Editura Teora, 2002.

EVALUATION		
	conditions	results corresponding to at least 80% of seminar/laboratory contact hours.
	criteria	For passing , at least a 6 grade is required for the seminar/laboratory tests.
	evaluation methods	Three tests: two tests for seminar/laboratory hours (during the semester; about 4 hours per week expected workload) and one final test from the course content (during examination weeks; about 1.5 hours per week expected workload during teaching period and 14 hours recap during examination period). In total, about 94 hours expected workload.
	final result - formula	Weighted average between the average mark of the two practical tests (weight 50%) and the mark for the final test (weight 50%).

COURSE NAME	**HARDWARE PRACTICE**	CODE: CS1211

STUDY YEAR	I	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
1	-	2	-	42	108	5	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	ENG. ADRIAN BUBURUZAN LECT. DR. VLAD RĂDULESCU	Computer Science

PREVIOUS COURSES REQUESTED	Computer Architecture and Operating Systems

OBJECTIVES	Understanding the hardware structure of the computers. Acquiring the knowledge regarding the hardware and software configuration of a PC.
GENERAL DESCRIPTION	Functional blocks of the PC. Microprocessors: internal structure, Intel and AMD platforms, techniques used for improving the performance. The memory (RAM, BIOS, video, cache). I/O devices: interrupts, buses, connecting standards. Setting up computer networks: network cards and standards, cables, hubs, switches, routers. Hard disk management: plugging, partitioning, formatting. CMOS configuration, connecting to the Internet, setup and configuration of a small PC network.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Assembling and configuring a computer, working with hard disks, building and configuring computer networks. Low-level programming.
TEACHING METHODS	Exposition, practical demonstrations.

BIBLIOGRAPHY (SELECTION)	
	J. L. Hennessy, D. A. Patterson, *Computer Architecture - A Quantitative Approach*, Morgan Kaufmann Publishers, 1990. D. A. Patterson, J. L. Hennessy, *Computer Organization & Design: The Hardware/Software Interface*, Morgan Kaufmann Publishers, 1998. A. Tanenbaum, *Structured Computer Organization*, Prentice Hall, 1999.

EVALUATION		
	conditions	The presence at the laboratoty activities.
	criteria	At least 5 points on each test; 4 points are accepted for at most one test.
	evaluation methods	Two practical tests, the first (TS1) during the 7th week, the second (TS2) during the 14th week. One written test (TL) during the examination session.
	final result - formula	Final result = (TS + TL1 + TL2) / 3

COURSE NAME	**ENGLISH**	CODE:CS1212

STUDY YEAR	I	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
0	2	0		28	122	5	M	ENGLISH

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	DRD. DRAGOS ZETU	Computer Science

PREVIOUS COURSES REQUESTED	

OBJECTIVES	To improve the student's proficiency in English in general, computer science English in particular.
GENERAL DESCRIPTION	The seminars will provide the student the possibility to work with authentic materials, interact and improve his/her grammar skills.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	The seminars will provide the student the possibility to work with authentic materials, interact and improve his/her grammar skills.
TEACHING METHODS	

BIBLIOGRAPHY (SELECTION)	Any English grammar compendium.

EVALUATION		
	conditions	Attendance, written exam
	criteria	Good proficiency in english
	evaluation methods	Continuous evaluation, written exam
	final result - formula	50% exam results, 50% seminar activity

COURSE NAME	**COMPUTER NETWORKS**	CODE: CS2101

STUDY YEAR	II	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	124	5	P	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	ASSOC.PROF. DR. SABIN-CORNELIU BURAGA	Computer Science

PREVIOUS COURSES REQUESTED	Operating Systems, Algorithms and Programming

OBJECTIVES	To provide a comprehensible vision of computer networks design issues. The students will be able to program network (Internet) applications based on TCP/IP protocols. Key information regarding different Internet paradigms will be also provided: client/server model, remote procedure call, and peer-to-peer model.
GENERAL DESCRIPTION	Core concepts. Terminology. Types of computer networks. Network design. ISO/OSI and TCP/IP stack of protocols. Medium access control. Ethernet. Network layer. IP protocol. ICMP, ARP, RARP. Routing. Transport layer. TCP and UDP protocols. Discussions. Programming network (Internet) applications. Client/server model. BSD socket API. Domain Name System (DNS). Application layer. Protocol design. Core Internet services. Terminal access, file transfer, e-mail. RPC (Remote Procedure Call) paradigm. Examples and practical deployment. Peer-to-peer model. Classification, aspects of interest, applications. Wireless networks. Computer network security.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Process communication on different machines. BSD socket API. Low-level primitives. Iterative and concurrent TCP and UDP applications. Socket options. Asynchronous communication. I/O multiplexing. Out-of-band data transmission.
TEACHING METHODS	Interactive presentations. Direct interaction. Online access to additional resources via the Website course.

BIBLIOGRAPHY (SELECTION)	1. S.Buraga, G.Ciobanu, *Atelier de programare în rețele de calculatoare* (in Romanian), Polirom, Iași, 2001. 2. S. Dixit, R. Prasad (eds.), *Wireless IP and Building the Mobile Internet*, Artech House, 2003. 3. A. Kshemkalyani, M. Singhal, *Distributed Computing. Principles, Algorithms, and Systems*, Cambridge University Press, 2008. 4. C. McNab, *Network Security Assessment*, O'Reilly, 2004. 5. R.Stevens, B.Fenner, A.Rudoff, *UNIX Network Programming Volume 1, Third Edition: The Sockets Networking API*, Addison Wesley, 2003. 6. A.Tanenbaum, *Computer Networks* (4th Edition), Addison-Wesley, 2002. 7. * * *, *IETF Request for Comments* (RFCs): http://www.ietf.org/rfc/

EVALUATION		
	conditions	1 project (P), 1 optional test during semester (T), lab assignments (L), other individual activities (A)
	criteria	project P>5, test T>5
	evaluation methods	1 project (P), 1 optional test during semester (T), lab assignments (L), other individual activities (A)
	final result - formula	0.4*P+0.3*1+0.1*L+0.1*A+1

COURSE NAME	**DATABASES**	CODE: CS2102

STUDY YEAR	II	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	94	5	P	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. VICTOR FELEA	Computer Science

PREVIOUS COURSES REQUESTED	-

OBJECTIVES	The presentation of some theoretical models for databases: relational, entity-relational. The implementation of these models and the realization of programming products in ORACLE.
GENERAL DESCRIPTION	The description of some theoretical models for databases: relational, entity-relational. Data structures for databases. Indexes, creation, states, updates, data described languages. Constraints in databases. Constraints of functional dependency type. Systems of formal rules for functional dependencies. The study of functional dependencies using propositional calculus. Constraints of multi-valued dependency type, systems of formal rules for functional and multi-valued dependencies. Implementation of constraints in some management systems for databases. Some aspects of implementation of relational models in management systems for databases. The design for databases schemes. Transactions in databases.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Laboratory works: the study of SQL -Oracle language. Realizing of themes using SQL – Oracle language. Elements of PL/SQL language. Analysis of real problems and realizing them in PL/SQL Oracle.
TEACHING METHODS	Presentation of the course content., verification of individual themes realized in Oracle.

BIBLIOGRAPHY (SELECTION)	1.Abiteboul S. etc:"Foundations of Databases", Addison Wesley, 1995. 2..Date C.J: Baze de date, traducere din engleză de Simona Preda şi Titi Preda, Editura Plus, 2005. 3.Date C.J.":An Introduction to Database Systems, ed.8, Addison Wesley, 2004. 4.Date C.J. :"Constraints and Predicates, A brief Tutorial", www.dbdebunk.com, 2001. 5.Fotache M. etc. :"Oracle 9i – Ghidul dezvoltarii aplicatiilor profesionale", Polirom, 2003. 6.Garcia-Molina H., Ullman J.D. :"Database Systems.The Complete Book", 2000. 7.D.Maier:"The Theory of Relational Databases, Academic Press, 1992. 8.Popescu Ileana: "Modelarea bazelor de date", Editura Tehnica, Bucuresti, 2000. 9.Rob P. etc.:"Database Systems Design, Implementation and Management", 95. 10.Felea V. :"Baze de date relationale. Dependente", Ed. Univ. Iasi, 95. 11.Felea V. :"Elemente ale implementarii modelului relational in sisteme de gestiune de baze de date. Ed.MatrixROM, 2007. 12.Felea V., Matei C. si Balta M.:"Interogarea bazelor de date. Aplicatii in Oracle si SQL Server", Ed.MatrixROM, 2005. 13.Documentatia produselor Oracle.

EVALUATION	conditions	Laboratory Activity (Lab), Written Tests (L1, L2)
	criteria	L1+L2>=20, Lab >=20 Max{L1}=30p, max{L2}=30p, max{Lab}=40p.
	evaluation methods	L1 in 7-th week, L2 in 14-th week Lab in laboratories.
	final result - formula	The marks depend on the percentages fixed by the university.

COURSE NAME	**FORMAL LANGUAGES, AUTOMATA AND COMPILERS**	CODE: CS2103

STUDY YEAR	III	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	1	1	-	56	94	5	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. GHEORGHE GRIGORAS	Computer Science

PREVIOUS COURSES REQUESTED	Algorithms and Programming (CS1101), Object - Oriented Programming (CS1207)

OBJECTIVES	Teaching fundamental concepts and results on the formal languages (especially those of type 2 and 3), finite automata and pushdown automata. Building a lexical analyser using regular expressions and a scanner generator e.g. Lex Building a sintactic analyser for an context free grammar using a parser generator e.g. Yacc Using Lex - Yacc for designing a interpreter / compiler for a programming language
GENERAL DESCRIPTION	Languages and grammars, grammars classification (Chomsky hierarchy), Regular grammars and languages, Finite automata and accepted languages, Equivalence of deterministic models with the nondeterministic ones and witth regular grammars, Context free grammars and languages, Derivations in context free grammars, Ambiguity, Recognition of context free languages, Removing epsilon rules, Removing rules of the form A->B, Chomsky normal form, Pushdown automata, Programming languages: design, implementation, Lexical analysis, Syntax analysis, Semantic analysis, Top down (predictive) syntax analysis, Bottom up (shift reduce) syntax analysis, Recursive descent syntax analysis, LL Syntax Analysis, LR Syntax Analysis, Translation to intermediate code.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Examples of languages and grammars, Deterministic finite automata, Nondeterministic automata, Epsilon-transition automata; example, Regular expressions, examples with reference to lexical units of programming languages, Context free grammars, the derivation of trees, eliminating unnecessary symbols, eliminating rules of erasing , Chomsky normal form, with word recognition algorithm CYK, Automatic Pushdown; examples, lexical analyzer's manual, obtained with an analyzer tool-LEX, analyze using tools YACC type. Interpreter built with Lex and YACC.
TEACHING METHODS	Slides with course items; seminar themes; projects' issues; electronic version of the course; main readings will be find on the web page

BIBLIOGRAPHY (SELECTION)	Grigoras, Gh. Constructia compilatoarelor - Algoritmi fundamentali, Ed. Universitatii Al. I. "Cuza Iasi", 274 pg., 2005. Jucan Toader - Limbaje formale şi automate, Editura Matrix Rom, Bucureşti, 1999, 162 p. Jucan Toader, Ştefan Andrei – Limbaje formale şi teoria automatelor. Teorie şi practică, Editura Universității "Al. I. Cuza", Iaşi, 2002, 327p. Stoughton Alley, Formal Language Theory, Kansas State University, Draft of Fall 2007. Yehezkael R.B., Course notes on Formal Languages and Compilers, Jerusalem College of Technology, December 2004. Internet resources: Manual LEX, Manual FLEX, Manual YACC, Manual Bison, Compiler Construction using Flex and Bison

EVALUATION		
	Conditions	Seminars' activity (SA), participation to tutorial hours for clarifying the issues regarding project elaboration, participation to laborator hours (LA), participation to final exam (FE)
	Criteria	SA >= 5, LA >= 5, FE >= 5
	Evaluation methods	Mixed (during the semester and examination)
	Final result – formula	Formula of the final score: 30% SA + 30% LA + 40% FE and ECTS criteria

COURSE NAME	**GRAPH ALGORITHMS**	CODE: CS2104

STUDY YEAR	II	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	2	-	-	56	124	5	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. CORNELIUS CROITORU	Computer Science

PREVIOUS COURSES REQUESTED	Algorithms and Programming (data structures) (CS1101)

OBJECTIVES	The students will be familiarised with the basic notions and results of the Algorithmic Graph Theory, which will be applied in the design of efficient algorithms for various combinatorial optimization problems.
GENERAL DESCRIPTION	Complexity Classes, Graph Theory vocabulary, Path problems (graph traversal, shortest paths, connectivity), Minimum spanning trees (union-find, amortized complexity), Matchings, Flows, Polinomial reductions for decision problems on graphs, Approaches for NP-hard problems on graphs, Planar Graphs.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Each seminary debates 4 problems (some of them, very difficult) in order to deepen the subjects introduced in the course. All problems are posted at the begining of the semester such that interested students could try to find original solutions or to search similar questions in the related bibliography.
TEACHING METHODS	Video presentations of the slides (containing the course notes) available in pdf format at the beginning of the semester (http://thor.info.uaic.ro/~croitoru/ag/ag08-09allinone.pdf).

BIBLIOGRAPHY (SELECTION)	
	CROITORU C., *Tehnici de baza in optimizarea combinatorie*, Editura Univ. Al. I. Cuza Iasi, Iasi,1992. CROITORU C., *Introducere in proiectarea algoritmilor paraleli*, Editura Matrix Rom, Bucuresti, 2002. TOMESCU I., *Probleme de combinatorica si teoria grafurilor*, Editura did. si ped., Bucuresti,1981. DIESTEL R., *Graph Theory*, Electronic Edition. CORMEN T.H., Leiserson C.E., Rivest R.L., Stein C., *Introduction to Algorithms*,MIT Press 2001.

EVALUATION		
	conditions	
	criteria	A student will be considered to have passed the exam if (s)he obtains at least 50 points.
	evaluation methods	-Seminary activity (attendance, work quality): : 0-18 points. - Homeworks (3 homeworks, in weeks 4, 8,12) each giving maximum 14 points: 0-42 points. - Written Final test : 0-60 points.
	final result - formula	The final grade (if the total number of points is at least 50) is given by applying the ECTS rules (adapted to FII).

COURSE NAME	**CODING THEORY AND CRYPTOGRAPHY**	CODE: CS2105O1

STUDY YEAR	II	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	OP

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	124	6	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. FERUCIO-LAURENȚIU ȚIPLEA	Computer Science

PREVIOUS COURSES REQUESTED	The minimal prerequisites for students taking this course are a rather elementary knowledge of linear algebra, number theory, algorithm design, and complexity. Students with more mathematical background and maturity will be able to move rather quickly through the material.

OBJECTIVES	The course is designed to introduce coding theory and cryptography to students in computer science, mathematics, and engineering.
GENERAL DESCRIPTION	The course is devided into two parts. The first part, coding theory, deals with variable length codes with application to data compression, and block codes with application to data communication over a noisy channel. The second part, cryptography, introduces several major themes predominate in the field: perfect secret, symmetric-key cryptography, public-key cryptography, hash functions, digital signatures, authentication, secret sharing, application to computer security (Kerberos, PGP, IP-security, electronic commerce etc.).
DESCRIPTION OF SEMINARY / LABORATORY WORK	Seminars and laborators are grouped around the chapter currently discused in the course. They are aimed to illustrate the topics of the chapter mainly by practical applications.
TEACHING METHODS	On-line and blackboard presentation.

BIBLIOGRAPHY (SELECTION)	• E. Kranakis. *Primality and Cryptography*, John Wiley & Sons, 1987. • A.J. Menezes, P.C. van Oorschot, S.A. Vanstone. *Handbook of Applied Cryptography*, CRC Press, third printing, 1997. • V.S. Pless, W.C. Huffman. *Handbook of Coding Theory*, Elsevier, 1998. • D. Salomon. *Data Compression. The Complete Reference*, Springer Verlag, 1998. • F.L. Tiplea. *Introduction to Coding Theory* (in preparation). • F.L. Tiplea. *Introduction to Cryptography* (in preparation).

EVALUATION	conditions	
	criteria	
	evaluation methods	6 homeworks and a final exam
	final result - formula	50% from the homeworks and 50% from the final exam

COURSE NAME	**CONTINUOUS MODELS AND MATLAB**	CODE: CS2105O2

STUDY YEAR	II	SEMESTER	1	COURSE STATUS (**C**-compulsory / **OP**-optional / **F**-facultative)	OP

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	2	-	-	56	94	5	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	LECT. DR. FLORIN IACOB	Computer Science

PREVIOUS COURSES REQUESTED	Mathematics; Algorithms and Computer Programming; e-Communication; English (I + II); Oriented - Object Programming.

OBJECTIVES	• The presentation of the basic concepts of mathematical modelling , mainly relied on differential equations and dynammical systems. • The knowledge about MATLAB and its applications in the study of continuous mathematical models.
GENERAL DESCRIPTION	Mathematical modelling (generalities, basic notions and principles). Discrete and continuous models (description, main characteristics). Differential equations and dynamical systems (basis). Some models governed by differential equations (examples). MATLAB (foundations; important tools; programming environment). The analysis of continuous models by MATLAB facilities.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Illustrated techniques of mathematical modelling. Simple, classic and modern, discrete and continuous models (concrete presentation). Differential equations and dynamical systems (by means of practical exercises). Certain models based on differential equations (examples). MATLAB elements (computer applications). Mathematical modelling using MATLAB (laboratory activities).
TEACHING METHODS	Theoretical exposition. Practical illustration. Heuristic procedures. Computerized analysis.

BIBLIOGRAPHY (SELECTION)	Isoc Dorin – *The Practice of Computerized Mathematical Modelling of Dynamical Systems* (in Romanian), Ed. Mediamira, Cluj-Napoca, 2001. Popa Marin – *The Basis of Computer Networks Modelling* (in Romanian), Ed. Univ. Bucureşti, 2004. F. Stănciulescu – *The Modelling of the Great Complexity Systems* (in Romanian), Ed. Tehnică, Bucureşti, 2003. A.C. Fowler – *Mathematical Models in the Applied Sciences*, Cambrigde Text in Applied Mathematics, Cambridge University Press, 1997. N. Gastinel – *Mathématique pour l'informatique: Équations différentielles*, Armand Colin séries, Paris, 1970. A. Dumitrescu – *MATLAB – Guide* (in Romanian), Ed. Teora, Bucureşti, 2001. D. Arnold, J.C. Polking – *Ordinary Differential Equations using MATLAB*, MathWorks (on line), 2008.

EVALUATION	conditions	Full participation at the laboratory works.
	criteria	Course and seminary/laboratory frequency; Active participation at **s**eminaries and laboratory classes; Homeworks preparation; Runing the written tests during the semester and the exam.
	evaluation methods	Marking by points the frequency at course (PCF), seminary (PSF) and laboratory (PLF) classes, the activity at seminaries (PAS) and laboratory works (PALW), the homeworks effectuation (PHE), the involving in the councelling programme (PICP) and the result at the two tests (TS1, TS2).
	final score (FS) – formula, examination condition; final mark establishing	FS = 60%*****PTS + 20%*****TS1 + 20%*****TS2, where PTS = PCF + PSF + PAS + PLF + PALW + PHE + PICP $\geq$ 30. Final mark is established using FS, by the new rule ECTS.

COURSE NAME	**GAME THEORY**	CODE: CS2105O3

STUDY YEAR	II	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	OP

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	2	-	-	56	94	5	P	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	ASSOC. PROF. DR. RODICA BRÂNZEI	Computer Science

PREVIOUS COURSES REQUESTED	Mathematics Probability theory and statistics

OBJECTIVES	This course is intended to provide a general insight in the field of game theory and its applications in real-life situations, economics, social sciences, computer science, operations research, etc. Game theory deals with mathematical models for competition and cooperation. The course is mainly aimed to enlighten the benefits achieved via interactions between game theory and computer science. This course is a must for students interested in (Master and) PhD programs and scientific research concerning (the interface between informatics and) game theory.
GENERAL DESCRIPTION	Introduction to game theory and its applications. Rational choice theory, attitudes towards risk, representation of information and uncertainty. Equilibrium computation and the complexity of finding Nash equilibria. Routing games, network formation games and the potential function method. Information and game theory: complete information versus incomplete information; perfect information versus imperfect information; static games versus dynamic games under different scenarios regarding information. Models in cooperative game theory, basic soluton concepts and related algorithms. Introduction to mechanism design (for computer scientists). Interaction between game theory and computer science (criptography and game theory; game theory and artificial intelligence; game theory and Internet protocols; game-theoretic aspects of computing).
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Representing a broad range of real-life situations (including computer science situations) and parlor games (including games that computers play) as non-cooperative or cooperative games. Solving different non-cooperative and cooperative games using traditional methods and available software. Analysing the complexity of algorihms for computing solutions of games. Designing algorithms for computing solutions of special classes of cooperative games.
TEACHING METHODS	Using overhead projector and blackboard.

BIBLIOGRAPHY (SELECTION)	
	1. R. Brânzei, Game Theory, Alexandru Ioan Cuza University Press, Iasi, 2006 (in Romanian). 2. R. Brânzei, D. Dimitrov, S. Tijs, Models in Cooperative Game Theory: Crisp, Fuzzy and Multi-choice Games, Springer-Verlag, Vol. 556, Berlin, 2005. 3. N. Nisan, T. Roughgarden, É. Tardos, V. Vazirani (Eds.), Algorithmic Game Theory, Cambridge University Press, 2007.

EVALUATION		
	conditions	ME (meadterm evaluation), FE (final evaluation), W (Seminary/Laboratory Works)
	criteria	ME ≥ 6, FE ≥ 4, W $\in \{0, 1, 2\}$
	evaluation methods	ME (written test (75 min) covers weeks 1-6) and F T (written test (75 min) covers weeks 8-13)
	final result - formula	ME + FE + W

COURSE NAME	**ENGLISH FOR COMPUTER SCIENCE**	COD: CS2106

STUDY YEAR	II	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

CLASSES PER WEEK				TOTAL CLASSES PER SEMESTER	TOTAL CLASSES INDIVIDUAL ACTIVITY	NUMBER OF CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
-	-	2	-	28	122	5	M	English

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	ASSIST. NICOLETA LEON	Computer Science

PREVIOUS COURSES REQUESTED	-

OBJECTIVES	Students will learn to use a language adequate to the Computer Science field. Students will gain the necessary skills to participate in a job interview, will be able to write a CV and a covering letter in English, etc.
GENERAL DESCRIPTION OF THE COURSE	Grammar: ”if” clauses, adjectives, sequence of tenses, prepositions. Vocabulary: specific to Computer Science + other fields of interest for the students. Topics of discussion: Artificial Intelligence, Cryptography, Windows vs Linux, Dependence on computers, etc.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	The students will present a project in order to prove their ability of using accurate English and of expressing their ideas by means of a vocabulary adequate to the themes they present.
TEACHING METHODS	Interactive methods used all along the seminar.

COMPULSORY BIBLIOGRAPHY (SELECTION)	

EVALUATION	conditions	Var 1.: Seminar activity Project presentation Var 2.: Project presentation Written exam
	criteria	Var 1.: Seminar minimum score: 50% of the maximum score that can be obtained at the seminar (minimum 50% of the score for the seminar homework and minimum 50 % of the score for the project) Var 2.: Seminar minimum score: 50% of the maximum score that can be obtained at the seminar (minimum 50% of the score for the seminar homework and minimum 50 % of the score for the project) Exam minimum score: 40% of the score that can be obtained at the exam
	evaluation methods	Seminar: seminar homework, project Written exam– work time: 1 hour
	final result formula	The final result is: Var 1.: a sum of the scores obtained for the seminar and for the project Var 2.: a sum of the scores obtained for the seminar, for the project and for the exam. The student that takes the exam will receive a grade, otherwise he/she will be considered absent. If one of the criterion of promotion is not fulfilled, the student will receive a grade equal to or lower than 4.

COURSE NAME	**WEB TEHNOLOGIES**	CODE:CS2207

STUDY YEAR	II	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	124	5	P	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	ASSOC. PROF. DR. SABIN-CORNELIU BURAGA	Computer Science

PREVIOUS COURSES REQUESTED	Computer Networks, Formal Languages, Automata and Compilers, Algorithms and Programming

OBJECTIVES	To study the actual methodologies, specifications and techniques of Web development and to offer the necessary skills for designing, implementing and deploying complex Web applications.
GENERAL DESCRIPTION	General architecture of the WWW space. Uniform resource identifiers (URIs). Hypertext. HTTP protocol. Extensible Markup Language (XML). XML family: namespaces, validation, transformation, processing methods. Web application programming. Common Gateway Interface (CGI) standard. Web application servers. Cookies and sessions. Architectures. Web services: SOAP, WSDL, UDDI. REST paradigm. Service oriented architecture (SOA). Social Web. Characteristics. Directions of evolution. Tagging. Syndication (RSS, Atom). Microformats. Advanced user-interaction (RIA, AJAX, Web widgets). Mash-up applications. Introduction to Web engineering. Web resource searching and retrieval technologies: robots, search engines, other approaches. Web application security.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	HyperText Markup Language (HTML). XHTML. CSS style sheets. Data modeling and processing in XML. Web programming – server side. Web application servers. Types of Web applications: from requirements to practical deployment.
TEACHING METHODS	Interactive presentations. Direct interaction. Online access to additional resources via the Website course.

BIBLIOGRAPHY (SELECTION)	
	1. L. Alboaie, S. Buraga, *Servicii Web* (in Romanian), Polirom, 2006. 2. M. Bowers, *Pro CSS and HTML Design Patterns*, Apress, 2007. 3. S. Buraga, *Tehnologii XML* (in Romanian), Polirom, 2006. 4. S. Buraga, *Proiectarea siturilor Web – ediția a doua* (in Romanian), Polirom, 2005. 5. S. Buraga (coord.), *Programarea în Web 2.0* (in Romanian), Polirom, 2007. 6. M. Cross *et al.*, *Web Application Vulnerabilities*, Syngress, 2007. 7. B. Daum, U. Merten, *System Architecture with XML*, Elsevier Science, 2003. 8. G. Kappel *et al.* (eds.), *Web Engineering*, John Wiley & Sons, 2006. 9. M. Zandstra, *PHP Objects, Patterns, and Practice* (2nd Edition), Apress, 2008. 10. * * *, *World Wide Web Consortium's Technical Reports*, 2008: http://www.w3.org/

EVALUATION		
	conditions	1 project (P), 1 optional test during semester (T), lab assignments (L), other individual activities (A)
	criteria	project P>5, test T>5
	evaluation methods	1 project (P), 1 optional test during semester (T), lab assignments (L), other individual activities (A)
	final result - formula	0.4*P+0.3*1+0.1*L+0.1*A+1

COURSE NAME	**ADVANCED TECHNIQUES OF PROGRAMMING**	CODE: CS2208

STUDY YEAR	II	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	154	5	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	LECT. DR. CRISTIAN FRĂSINARU	Computer Science

PREVIOUS COURSES REQUESTED	Object-oriented programming

OBJECTIVES	Introducing Java programming language and J2SE platform technologies. Description of various advanced programming techniques and modalities of implementig them using Java Introducing J2ME and presentation of fundamental notions concerning mobile devices programming. Presentation Sun J2SDK 1.6 development kit. Presentation of various integrated development editors (IDE) for Java (NetBeans, Eclipse).
GENERAL DESCRIPTION	Introduction. Objects and Classes. Exceptions. Data Streams. Interfaces. Packages. Collections. AWT. Swing. Threads. Networking. Applets. JDBC. Reflection. Java Web Start. Annotations. RMI. Introduction in Java ME. Introduction in Java EE.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Problems concerning each individual course.
TEACHING METHODS	Videoprojection

BIBLIOGRAPHY (SELECTION)	Cristian Frasinaru: *Practical Course in Java*, Matrix Rom Bucuresti (2005), ISBN 973-685-856-1

EVALUATION	conditions	Each laboratoy will conain two problems, each counted with 1 point. The exam will contain 20 questions, each counted with 1 point. Supplimentary work mai count additional points.
	criteria	To enter exam, each student must have 8 points. To pass exam, each student must have 5 points.
	evaluation methods	Problem presentation (during semester) and Exam (in session)
	final result - formula	Gauss curve on the total number of points 5%=10, 10%=9, 20%=8, 30%=7, 25%=6, 10%=5

COURSE NAME	**SOFTWARE ENGINEERING**	CODE: CS2209

STUDY YEAR	II	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	94	5	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	ASSIST. DR. ADRIAN IFTENE	Computer Science

PREVIOUS COURSES REQUESTED	**Compulsory:** Algorithms and Programming. Object - Oriented Programming. **Recommended:** Computer Architecture and Operating Systems. Operating Systems. Advanced Techniques of Programming. Databases. Computer Networks.

OBJECTIVES	Building a professional attitude towards software development. Students learn advanced methods and techniques enabling them to develop quality software artifacts while observing the customer's demands for functionality, costs and deadline.
GENERAL DESCRIPTION	Software process models. Requirements engineering. Modeling and modeling languages. UML. Design patterns. Functional testing. Structural testing. Software metrics. Project management. Ethics.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	The students must implement a project with a medium complexity, following the steps presented at courses: Requirements engineering, Building of UML diagrams, Using of Design patterns, Testing of the Project using manual and automatic testing, Project evaluation using Software metrics.
TEACHING METHODS	Course slide presentations. Lecture notes and tutorials available electronically.

BIBLIOGRAPHY (SELECTION)	- Ian Sommerville: *Software Engineering*, Addison Wesley, 2001 - Craig Larman: *Applying UML and Patterns*, Addisson Wesley, 2002 - Erich Gamma, Richard Helm, Ralph Johnson, John Vissides: *Design Patterns, Elements of Reusable Object-Oriented Software*, Addisson Wesley, 1998

EVALUATION		
	conditions	Lab activity Project development Exam
	criteria	Minimum lab score 50% from maximum lab score (minimum 50% from labs tasks and minimum 50% from project score) Minimum exam score 40% from maximum exam score
	evaluation methods	Lab: weekly tasks, project Written exam – time for exam 30 minutes
	final result - formula	Final score is the sum of lab score, project score and exam score For the students who satisfy the evaluation criteria, the final grade is obtained by applying the Gauss curve on the final scores: grade 10 – first 5% grade 9 – next 10% grade 8 - next 20% grade 7 – next 30% grade 6 – next 25% grade 5 – last 10% The student who takes part in the exam receives a grade, otherwise he is considered absent. If any of the evaluation criteria are not met, the student receives a grade lesser than or equal to 4.

COURSE NAME	**DBMS PRACTICE**	CODE: CS2210

STUDY YEAR	II	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
1	-	2	-	42	94	5	P	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. VICTOR FELEA	Computer Science

PREVIOUS COURSES REQUESTED	-

OBJECTIVES	Expression of complex queries using the language of SELECT commands. The Design of database Schemes for real applications.
GENERAL DESCRIPTION	The expression of complex queries using SELECT commands, the processing of trees using SELECT- SQL Oracle command, general languages for database queries, running plans for realizing of queries, design algorithms for database schemes.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Laboratory works: The expression of complex queries for some real queries. The realizing of a project that contains complex queries, the design of database schemes, indexes, using ORACLE management system for database.
TEACHING METHODS	Presentation of the course content, verification of individual projects realized in Oracle.

BIBLIOGRAPHY (SELECTION)	1.Abiteboul S. etc:"Foundations of Databases", Addison Wesley, 95. 2..Date C.J: Baze de date, traducere din engleză de Simona Preda şi Titi Preda, Editura Plus, 2005. 3.Date C.J.":An Introduction to Database Systems, ed.8, Addison Wesley, 2004. 4.Date C.J. :"Constraints and Predicates, A brief Tutorial", www.dbdebunk.com, 2001. 5.Fotache M. etc. :"Oracle 9i – Ghidul dezvoltarii aplicatiilor profesionale", Polirom, 2003. 6.Garcia-Molina H., Ullman J.D. :"Database Systems.The Complete Book", 2000. 7.D.Maier:"The Theory of Relational Databases, Academic Press, 1992. 8.Popescu Ileana: "Modelarea bazelor de date", Editura Tehnica, Bucuresti, 2000. 9.Rob P. etc.:"Database Systems Design, Implementation and Management", 95. 10.Felea V. :"Baze de date relationale. Dependente", Ed. Univ. Iasi, 95. 11.Felea V. :"Elemente ale implementarii modelului relational in sisteme de gestiune de baze de date. Ed.MatrixROM, 2007. 12.Felea V., Matei C. si Balta M.:"Interogarea bazelor de date. Aplicatii in Oracle si SQL Server", Ed.MatrixROM, 2005. 13.Documentatia produselor Oracle.

EVALUATION		
	conditions	Laboratory Activity (Lab), Written Test (L)
	criteria	L>=20, Lab >=20 Max{L}=50p, max{Lab}=50p.
	evaluation methods	L in 7-th week, Lab in laboratories.
	final result - formula	The marks depend on the percentages fixed by the university.

COURSE NAME	**LOGIC PROGRAMMING**	CODE: CS2211O1

STUDY YEAR	II	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	OP

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	64	4	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	DRD. ALAIBA VASILE	Computer Science

PREVIOUS COURSES REQUESTED	Logics for Computer Science

OBJECTIVES	1. Getting minimal skills for programming in "PROLOG-based" computer languages. 2. Learning an existing language implementation.
TEMATICĂ GENERALĂ	1. First-order predicate logic and resolution in FOL (LP1) 2. Horn formulae and (definite) logic programs. 3. Refinements (restrictions and strategies) of resolution. 4. Semantics o f Logic programming. Complete canonical computation trees. 5. Non-determinism and negation. 6. The construction of a PROLOG-like interpreter. 7. Alternate Logic Programming paradigms
DESCRIPTION OF SEMINARY / LABORATORY WORKS	The themes will be fixed by the teacher at the beginning of the semester, according to the specified context. They will include the knoledge verification for the previous courses, short examples to be programmed and executed and larger projects.
TEACHING METHODS	All the classical didactic methods will be used: systematic exposure of knoledge, conversation, learning „by descovery", etc. The Courses will be taught using a retro- or video-porojector. A recent language implementation, SWI-PROLOG will be taught at the laboratory classes.

BIBLIOGRAPHY (SELECTION)	4. C.D. Masalagiu – Fundamentele logice ale Informaticii, Editura Universității „Al. I. Cuza", Iași, 2004, ISBN 973-703-015-X (in Romanian). 5. C. Cazacu, V. Slabu – Logică Matematică, Editura „Ștefan Lupașcu", Iași, 1999, ISBN 973-99044-0-8 (in Romanian). 6. V. Cotelea – Programarea în logică, Editura „Nestor", Chișinău, Republica Moldova, ISBN 9975-9606-0-X (in Romanian). 7. J. W. Lloyd – Foundations of Logic Programming, Spriger Verlag, Germany, 1984, ISBN 3-540-13299-6. 8. Jan Wielemaker, SWI-Prolog 5.6 Reference Manual, Human-Computer Studies, University of Amsterdam, updated March 2007 9. U. Nilsson, J. Maluszynski, Logic, Programming and Prolog (2ed), Wiley 1995, online edition, 2000. 10. R. Kowalski, Algorithm = Logic + Control, Communications of the ACM, Volume 22, Issue 7 (July 1979) 11. P.Blackburn, J.Bos, K.Striegnitz, Learn Prolog Now!, Texts in Computer Science, vol.7, College Publications, 2006 12. J.R.Fisher, prolog :- tutorial, online version, updated January 2007. 13. L.Sterling, E.Shapiro, The Art of Prolog, Second Edition: Advanced Programming Techniques (Logic Programming), The MIT Press, 1994.

EVALUATION		
	conditions	Attendance is mandatory at the practical laboratories.
	criteria	The on-line tests during the normal practical activity (laboratory) may generate 90 points (maximum, see below). To "graduate" the course, a minimum of 50 points is need.
	evaluation methods	Concerning the "lab". During the semester activity, there will be 5 themes (each quoted at 10 points, maximum) and a more complex project (quoted at 40 points, maximum).
	final result - formula	The final grade is computed by first summing up all the obtained points and applying a Gauss-like distribution, according to existing reglementations.

COURSE NAME	**FUNCTIONAL PROGRAMMING**	CODE: CS2211O2

STUDY YEAR	II	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	OP

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	94	5	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. GHEORGHE GRIGORAS	Computer Science

PREVIOUS COURSES REQUESTED	

OBJECTIVES	Understanding the concepts of functional programming, ability and skills in programming language Haskell. The concepts presented will include: functions, types, lists, abstract data types, high-order functions, side effect, lazy evaluation. Students will use language Haskel for implementation of data types and algorithms for solving problems.
GENERAL DESCRIPTION	Introduction to functional programming, The Hugs system, Introduction to the language Haskel, types and classes, polymorphism, recursive function, lists, trees, high-order functions, lazy evaluation, monad, definition of new types, Case Studies.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Hugs system, Programming in Haskell, definition of functions, recursive functions and high-order functions, interactive programs, 2 individual projects.
TEACHING METHODS	Slides with course items; seminar themes; projects' issues; electronic version of the course; main readings will be find on the web page

BIBLIOGRAPHY (SELECTION)	Richard Bird: Introduction to Functional Programming using Haskell, Prentice Hall, 1998. Graham Huton, Programming in Haskell, http://www.cs.nott.ac.uk/~gmh/. Mihai Gontineac, Programare functionala - O introducere utilizand limbajul Haskell, Ed. Al Myller Iasi, 2006 Limbajul Haskell: www.haskell.com. Ro/Haskell: http://www.haskell.org/haskellwiki/Ro/Haskell

EVALUATION	Conditions	laborator activity (LA), final exam (FE)
	Criteria	LA >= 5, FE >= 5
	Evaluation methods	Mixed (during the semester and examination)
	Final result – formula	Formula of the final score: 50% LA + 50% FE and ECTS criteria

COURSE NAME	**ENGLISH FOR COMPUTER SCIENCE**	COD: CS2212

STUDY YEAR	II	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

CLASSES PER WEEK				TOTAL CLASSES PER SEMESTER	TOTAL CLASSES INDIVIDUAL ACTIVITY	NUMBER OF CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
-	-	2	-	28	122	5	M	English

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	ASSIST. NICOLETA LEON	Computer Science

PREVIOUS COURSES REQUESTED	-

OBJECTIVES	Students will learn to use a language adequate to the Computer Science field. Students will gain the necessary skills to participate in a job interview, will be able to write a CV and a covering letter in English, etc.
GENERAL DESCRIPTION OF THE COURSE	Grammar: ”if” clauses, adjectives, sequence of tenses, prepositions. Vocabulary: specific to Computer Science + other fields of interest for the students. Topics of discussion: Artificial Intelligence, Cryptography, Windows vs Linux, Dependence on computers, etc.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	The students will present a project in order to prove their ability of using accurate English and of expressing their ideas by means of a vocabulary adequate to the themes they present.
TEACHING METHODS	Interactive methods used all along the seminar.

COMPULSORY BIBLIOGRAPHY (SELECTION)	

EVALUATION	conditions	Var 1.: Seminar activity Project presentation Var 2.: Project presentation Written exam
	criteria	Var 1.: Seminar minimum score: 50% of the maximum score that can be obtained at the seminar (minimum 50% of the score for the seminar homework and minimum 50 % of the score for the project) Var 2.: Seminar minimum score: 50% of the maximum score that can be obtained at the seminar (minimum 50% of the score for the seminar homework and minimum 50 % of the score for the project) Exam minimum score: 40% of the score that can be obtained at the exam
	evaluation methods	Seminar: seminar homework, project Written exam– work time: 1 hour
	final result formula	The final result is: Var 1.: a sum of the scores obtained for the seminar and for the project Var 2.: a sum of the scores obtained for the seminar, for the project and for the exam. The student that takes the exam will receive a grade, otherwise he/she will be considered absent. If one of the criterion of promotion is not fulfilled, the student will receive a grade equal to or lower than 4.

COURSE NAME	**DESIGN AND ANALYSIS OF ALGORITHMS**	CODE: CS3101

STUDY YEAR	III	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	1	1	-	56	94	5	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF.DR. DOREL LUCANU	Computer Science

PREVIOUS COURSES REQUESTED	Algorithms and Programming (CS1101)

OBJECTIVES	Understanding algorithm analysis, algorithm design techniques, handling intractable problems.
GENERAL DESCRIPTION	Advanced data structures, design paradigms (greedy, dynamic programming, divide-et-impera, backtracking, branch-and-bound), NP-complete problems, modern heuristics.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Exercises helping to understand the algorithms, problem solving.
TEACHING METHODS	Slide-based intercative presentations, problem solving.

BIBLIOGRAPHY (SELECTION)	Dorel Lucanu, Mitica Craus. Proiectarea algoritmilor. Polirom, 2008. T.H. Cormen, C.E. Leiserson, R.L. Rivest: Introduction to Algorithms, MIT Press, 1990. T.H. Cormen, C.E. Leiserson, R.L. Rivest: Introducere in Algoritmi,Computer Libris Agora, 2000.

EVALUATION		
	Conditions	Seminar activity (SA), home-works (HW), Final written test (WT)
	Criteria	SA >= 5, LA >= 5, FE >= 5
	Evaluation methods	Mixed
	Final result – formula	10% AS +40%HW + 50% WT + Bonus for extra activity

COURSE NAME	**INFORMATION SECURITY**	CODE: CS3102

STUDY YEAR	III	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	94	5	M	English/Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	FACULTY/DEPARTMENT
	PROF. DR. FERUCIO LAURENȚIU ȚIPLEA	Department of Computer Science

PREVIOUS COURSES REQUESTED	No prerequisite required.

OBJECTIVES	This course is an advanced introduction to the theory of information and computer security. Students will gain basic and advanced knowledge about secure communications in the business world (e-commerce, e-payment, e-lotteries, e-gambling etc.) and how to secure their computers against electronic attacks.
GENERAL DESCRIPTION	The course includes: • brief introduction to cryptographic primitives (cryptosystem, digital signature, hash function, PKI); • security protocols and policies, models of security; • protocols for e-commerce, e-payment, identification and authentication, password management; • smart-card tehnology; • network security (IPsec, SSL, Kerberos, VPN, wireless, firewall); • system security; • e-mail security; • security of mobile devices.
DESCRIPTION OF SEMINARY / LABORATORY WORK	Seminars and laborators are grouped around the chapter currently discused in the course. They aim to illustrate the topics of the chapter mainly by practical applications.
TEACHING METHODS	On-line and blackboard presentation.

BIBLIOGRAPHY (SELECTION)	• William Stallings: Cryptography and Network Security: Principles and Practice, 4th Ed., Prentice Hall, 2005. • PGP, Ipsec, SSL, Kerberos etc. documentation.

EVALUATION	conditions	
	criteria	
	evaluation methods	7 homeworks and a final exam
	final result - formula	50% from the homeworks and 50% from the final exam

COURSE NAME	**ARTIFICIAL INTELLIGENCE**	CODE: CS3103

STUDY YEAR	III	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	94	5	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. DAN CRISTEA	Computer Science

PREVIOUS COURSES REQUESTED	-

OBJECTIVES	Introducing fundamental notions of the domain of Artificial Intelligence, beginning with the language Lisp, then presenting notions of knowledge representation, search in the state space, inferences in semantic networks and problems of planning and games.
GENERAL DESCRIPTION	**Chapter I Introduction**: Definition of the domain of AI, the Turing test, philosophical problems, sub-domains of AI **Chapter II Lisp**: Pure Lisp, s-expressions, syntax, representations of lists, evaluation of expressions, functions and macros, variables and their domains, closures, transfer of arguments in functions, recursivity **Chapter III Representation of knowledge and reasoning**: Human-environment interactivity, descriptive semantic networks (queries, demons, the system IURES), event-oriented semantic networks **Chapter IV Production systems**: Modelling AI problems, control, searching for solutions in state spaces, irrevocable strategies (hill-climbing), tentative strategies (backtracking hill-climbing), systematic search (depth-first search, breadth-first search, best-first), cost-guided search in graphs **Chapter V Planning and games**: Search in game trees, the MIN-MAX method, the alpha-beta method, robot planning, STRIPS rules, bringing the robot back in the plan
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Follows the themes taught at the course. Labs are posted at the address http://thor.info.uaic.ro/~orar/profesori/orar_ionita.html
TEACHING METHODS	Power Point presentations and interactive Lisp sessions

BIBLIOGRAPHY (SELECTION)	D. Cristea, I. Pistol, M. Ioniță: Artificial Intelligence (in Romanian), University Al. I. Cuza Publishing House, Iaşi, 2007 On-line courses posted at http://thor.info.uaic.ro/~dcristea/teaching.html

EVALUATION	conditions	Minimal request to enter the exam: 26 lab points (out of 36) + 50 project points (out of 100) Minimum to pass: 26 lab points + 50 project points + 50 written exam points (out of 100)
	criteria	Labs: 12*{● in the class, ●● solved exercise, ●●● remarked} ➔ max 36; project: 0 – 100; written exam: 0 – 100
	evaluation methods	Lab, project, written exam
	final result - formula	(1,1*100/36*lab+1,2*pro+ex)/30, modified by Gauss

COURSE NAME	**APPLICATIONS DEVELOPMENT USING .NET FRAMEWORK**	CODE: CS3104

STUDY YEAR	III	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	94	5	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	INF. IOAN ASIMINOAEI	Computer Science

PREVIOUS COURSES REQUESTED	Algorithms and Programming (CS1101)

OBJECTIVES	Teaching fundamental concepts in object-oriented programming and the use of the programming language C# on .NET Framework.
GENERAL DESCRIPTION	.NET Architecture. Value and Reference Type. Attribute and Interfaces. Garbage Collector and Exceptions. Events. Delegates. Publish-Subscribe pattern. Thread and Sincronization. LINQ. .NET Remoting. Web Services. Windows Communication Foundation.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Console, Windows and Library Applications. Collections classes. Threading and Sincronization. .NET Remoting, WCF, Windows and Web Services.
TEACHING METHODS	Slides with course items; seminar themes; projects' issues; electronic version of the course; main readings will be find on the web page

BIBLIOGRAPHY (SELECTION)	Tom Archer: Inside C# Second Edition. Scott McLean, James Naftel, Kim Williams: Microsoft .NET Remoting, 2002 J. Richter: Applied Microsoft .NET Framework Programming, 2002 Andrew Troelsen: Pro C# 2008 .NET 3.5 Platform Chris Sells, Michael Weinhardt: Windows Forms 2.0 Programming MSDN

EVALUATION	Conditions	Participation to laborator hours (LA), participation to writing tests (WT)
	Criteria	LA >= 6, WT >= 4
	Evaluation methods	Mixed (during the semester and examination)
	Final result – formula	60% LA + 40% WT

COURSE NAME	**STOCHASTIC PROCESSES**	CODE: CS3105O1

STUDY YEAR	III	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	OP

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	2	-	-	56	94	5	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	LECT. DR. ANCA VITCU	Computer Science

PREVIOUS COURSES REQUESTED	Mathematics, Theory of Probability and Statistics, Graphs

OBJECTIVES	The main aim of the course is to provide grounding in stochastic processes and their application in various fields like economy, finance, biology, medicine, genetics, and physics. It is a useful tool for: (i) understanding the models of some phenomena which have a random development as well as the hypothesis on which these models are designed; (ii) developing models in C++/JAVA. **On completion of the course the students will be able to**: (i) describe the principles of modeling; (ii) describe the general principles of stochastic processes and their classification into different types; (iii) define and apply a Markov chain and a Markov process; (iv) define and apply the main concepts of Gauss-Wiener processes and other Lévy processes; (v) explain and apply the concepts of "Monte Carlo" simulation of a stochastic process.
GENERAL DESCRIPTION	I. Preliminaries (Concepts of probability theory, Stochastic Processes – Markov chain, Markov processes, Poisson processes, Levy processes, Brownian Motion, Conditional Expectation , Martingales); II. Stohastic Integral (Riemann şi Riemann-Stieltjes Integrals, The Ito Integral, Ito Lemma, Stratonovich Integral and other types of integrals); III. Markov Processes and Diferential Stochastic Equations (Deterministic Differential Equations, Stochastic differential Equations, The Markov Property); IV. Girsanov Theorem; V. Martingale Representation Theorem
DESCRIPTION OF SEMINARY / LABORATORY WORKS	*Science* : Models of genetic population (i.e. Wright Model, Feller Model) ; *Finance*: introduction to basic financial concepts which will be used during course and seminars development: (- types of contracts and ways they are used on financial market: *working hypothesis of financial market*, *examples* – types of options (European, American, Asian, Exotic, Bermudan, Russian, Parisian) which development are important challenges for mathematicians, computer scienc experts and statisticians; types of contracts (forwards and futures), *types of participants* (hedgers, speculators, arbitrageurs, bulls and bears) ; - Basic hypothesis for models development (i.e. arbitrage free) ;- Binomial Models; - Modele utilizate la bursă (Wall Street) : Cox-Ross-Rubinstein (CRR) Model, Hull and White Model, Cox-Ingersoll-Ross Model, Brace-Gatarek-Musiela Model, Change of numeraire, Merton formula; *Simulation techniques*: Monte Carlo Method (describe how apparently pseudo-random integers can be generated using a computer; describe how pseudo-random drawings from specified distributions can be generated; explain how a series of sets of correlated normal random variates can be generated; explain the disadvantages of using random as opposed to pseudo-random numbers; discuss how to decide how many simulations to carry out for any particular purpose); *Implementation of other programming techniques*: (JAVA/C++) for some of the most important models used on the Wall Street (Hull-White, CIR, HJM, etc.)
TEACHING METHODS	Slides with course items; seminar themes; projects' issues; electronic version of the course; main readings will be find on the web page

BIBLIOGRAPHY (SELECTION)	[1] Baxter, M. and Rennie, A., (1996). Financial calculus. Cambridge University Press. [2] Campbell, J.Y., Lo, A.W. and MacKinlay, A.C., (1997). The Econometrics of Financial Markets. Princeton University Press. [3] Etheridge, A., (2002). A Course in Financial Calculus. Cambridge University Press. [4] Glasserman, P., (2004). Monte Carlo Methods in Financial Engineering, Springer. [5] Hunt, P.J. and Kennedy, J.E., (1998). Financial Engineering. Wiley. [6] Jorion, P., (2001). Value at Risk the New Benchmark for Managing Financial Risk. McGraw-Hill. [7] Kimmel, M., Axelrod D. E., (2002). Branching Processes in Biology. Springer. [8] Musiela, M. and Rutkowski, M., (1997). Martingale Methods in Financial Modelling. Springer-Verlag, Berlin. [9] Shreve, Steven E. (2004). Stochastic Calculus for Finance, Springer [10] Smithson, C.W., Smith, C.W. and Wilford, D.S., (1995). Managing Financial Risk. Irwin, Burr Ridge, Illinois.

EVALUATION		
	Conditions	Seminars' activity, participation to tutorial hours for clarifying the issues regarding project elaboration
	Criteria	Seminar work (five sets of problems to work for understanding), a project (prepared by a team composed of max 2 students and supervised by the professor in charge)
	Evaluation methods	Mixed (during the semester and examination)
	Final result – formula	70% evaluation during semester (30% seminar work + 40% for the understanding of the issues associated with the project), 30% final exam (project evaluation)

COURSE NAME	**COMPUTABILITY, DECIDABILITY, AND COMPLEXITY**	CODE: CS3105O2

STUDY YEAR	III	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	OP

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	94	5	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. FERUCIO LAURENȚIU ȚIPLEA	Computer Science

PREVIOUS COURSES REQUESTED	No prerequisite required.

OBJECTIVES	This course is an advanced introduction to theoretical computer science emphasizing two interrelated areas: the *theory of computability* (how to tell whether problems are algorithmically solvable) and the *theory of complexity* (how efficient an algorithm is).
GENERAL DESCRIPTION	The heart of the course material on computability is the recursion theory (primitive recursion and mu-recursion). The connection with Turing machines and WHILE-programs is made, and some important decision problems are discussed. The complexity theory is considered both from qualitative and quantitative points of view: dynamic complexity measures are introduced, the concept of a complexity class as a class of languages is presented, the strict correspondence between such classes and decision problems is established, and techniques used to study properties of such classes are formulated. Then, we study in details the basic properties of some of the most significant complexity classes. Finally, we deal with probabilistic algorithms and with the corresponding complexity classes.
DESCRIPTION OF SEMINARY / LABORATORY WORK	Seminars and laborators are grouped around the chapter currently discused in the course. They are aimed to illustrate the topics of the chapter mainly by practical applications.
TEACHING METHODS	On-line and blackboard presentation.

BIBLIOGRAPHY (SELECTION)	• J.L. Balcazar, J. Diaz, J. Gabarro. *Structural Complexity*, Vol I-II, Springer-Verlag, 1995. • M.D. Davis, R. Sigal and E.J. Weyuker: *Computability, Complexity and Languages*, 2nd Ed., Academic Press (Morgan Kaufmann), 1994. • J.E. Hopcroft, R. Motwani and J.D. Ullman: *Introduction to Automata Theory, Languages and Computation*, 2nd Ed., Addison-Wesley, 2001. • N.D. Jones. *Computability and Complexity*, MIT Press, 1997. • Ch.H. Papadimitriou. *Computational Complexity*, Addison-Wesley, 1994. • Journal papers.

EVALUATION		
	conditions	
	criteria	
	evaluation methods	6 homeworks and a final exam
	final result - formula	50% from the homeworks and 50% from the final exam

COURSE NAME	**NUMERICAL CALCULUS**	CODE: CS3207

STUDY YEAR	III	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY *	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	94	5	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	LECT. DR. ANCA IGNAT	Computer Science

PREVIOUS COURSES REQUESTED	Mathematics, Algorithms and Programming, Object-Oriented Programming

OBJECTIVES	Learning about numerical methods for approximation of continous mathematical problems.
GENERAL DESCRIPTION	• Examples, floating point computing, types of errors, propagation of errors • LU decompositions (Gauss elimination algorithm, Cholesky factorisation), QR decompositiond (Givens and Householder algorithms), singular value decomposition • Iterative methods for solving linear systems (Jacobi and Gauss-Seidel methods, succesive overrealaxation) • Eigenvalues and eigenvectors approximation (Jacobi method for symmetric matrices, QR type algorithms) • Solving nonlinear equations and systems of nonlinear equations (Newton type methods, false position method, secant method, methods for the roots of polynomials) • Polynomial interpolation (Lagrange polynomial, Newton polynomials), spline interpolation (linear continuous, cubic of class C^2) • Numerical integration (Newton-Cotes type formulae)
DESCRIPTION OF SEMINARY / LABORATORY WORKS	• Evaluation of elementary functions (sin/cos/...), errors in numerical computations; • Solving linear systems: 1. Substitution method, LU decomposition; 2. QR decomposition: Givens or Householder algorithm; 3. Iterative methods: Jacobi and Gauss-Seidel methods; • Jacobi method for finding the eigenvalues and eigenvectors for symmetric matrices; • Solving nonlinear equations: bisection method, Newton-Raphson method,false position method, secant method, methods for approximating roots of polynomials; • Polynomial interpolation: Newton-Lagrange polynomial, Aitken algorithm, C^2-cubic spline functions; • Numerical integration: Newton-Cotes type formulae, iterate methods.
TEACHING METHODS	Course – using videoprojector, Laboratory works - files describing the algorithms

BIBLIOGRAPHY (SELECTION)	
	• C. Ignat, C. Ilioi, T. Jucan, *Elemente de informatică şi calcul numeric*, Editura Univ. „Al.I. Cuza" Iaşi, 1989, • T.A. Beu, *Calcul numeric în C*, Editura Albastră, Cluj, 2000, • V. Iorga, B. Jora, *Metode numerice*, Ed. Albastra, Cluj, 2004 • S. Salleh, A.Y. Zomaya, S.A. Bakar, *Computing for Numerical Methods using Visual C++*, Wiley-Interscience, 2008

EVALUATION		
	conditions	- Implementing the homeworks - Final score (lab + exam) must exceed a certain threshold
	criteria	- Each homework (there are at most 8 homeworks) has a maximal score associated if the deadline is respected (the homeworks can be presented after the deadline but with penalties) - The exam consists of exercises with all resources available (evaluation between 1 - 10)
	evaluation methods	• Written exam • Evaluation of the implemented homeworks
	final result - formula	acording to the present regulations applied to the final score (laboratory score + exam score)

COURSE NAME	**COMPUTER GRAPHICS**	CODE: CS3208

STUDY YEAR	III	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	94	5	E	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	LECT. DR. LUCIAN GHIRVU	Computer Science

PREVIOUS COURSES REQUESTED	Algorithms and Programming, Object Oriented Programming

OBJECTIVES	Introduction to computer graphics. Acquiring the ability of designing simple models (i.e., a collection of statically or dynamically objects having simple geometric shapes. Acquiring the techniques of rendering models (by using raster graphics). Acquiring the ability of designing graphics software using a standard graphics API.
GENERAL DESCRIPTION	1. INTRODUCTION TO COMPUTER GRAPHICS. 2. USE OF COLOR IN COMPUTER GRAPHICS. 3. 2D/3D GEOMETRIC TRANSFORMATIONS AND THEIR REPRESENTATION BY MATRICES. 4. RASTER GRAPHICS, CLIPPING, ANTI-ALIASING. 5. 3D VIEWING TRANSFORMATIONS. 6. PARAMETRIC POLYNOMIAL CURVES AND SURFACES. 7. SPATIAL SUBDIVISION TECHNIQUES. 8. HIDDEN SURFACE REMOVAL. 9. REFLECTION AND SHADING MODELS. 10. TEXTURES. 11. ADVANCED MODELING TECHNIQUES: FRACTALS, ANIMATION TECHNIQUES. 12. OPENGL LIBRARY.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	1. OPENGL LIBRARY (AND GLUT TOOL). INTRODUCTION. USING OPENGL FOR 2D CURVES DRAWING. 2. COLOR. 3. GEOMETRIC TRANSFORMATIONS IN OPENGL. 4. 2D RASTER GRAPHICS. 5. 3D VIEWING TRANSFORMATIONS IN OPENGL. 6. REFLECTION AND SHADING MODELS, TEXTURES IN OPENGL. 7. PARAMETRIC POLYNOMIAL CURVES AND SURFACES.
TEACHING METHODS	Lectures are given using MS Office Powerpoint and blackboard presentations. During the laboratory classes the students will receive a set of solved problems (usually programming assignments) and they are asked to solve a set of similar problems or to implement (using OpenGL and Visual Studio 6.0) some algorithms schematically presented during the lectures.

BIBLIOGRAPHY (SELECTION)	1. F.Ionescu, Grafica în realitatea virtuală, Ed.Tehnică 2000. 2. M.Vlada, I.Nistor, A.Posea, C.Constantinescu, Grafică pe calculator în limbajele Pascal şi C, Ed. Tehnica 1991. 3. C.-D.Neagu, S.Bumbaru, Sisteme multimedia - Grafică pe calculator, Ed. Matrix Rom, 2001. 4. D.Hearn, M.P.Baker, Computer Graphics, C Version (2nd Edition), Prentice Hall 1996. 5. L.Raicu, Grafic şi vizual între clasic şi modern, Ed. Paideia, 2000. 6. F.Moldoveanu, Grafică pe calculator, Ed. Teora, 1996.

EVALUATION	Conditions	No absence at laboratory classes. In order to pass the exam the students are required to solve completely at least 25% of homeworks and to solve completely at least 25% of the written test.
	Criteria	Activity at laboratory classes (pertinent questions regarding the assignments, presentation of several stages of solving the assignment), strong lectures attendance (in case of low levels of attendance, the attendance of a particular student could be recompensed by granting bonuses to his/her final grade).
	Evaluation methods	Programming assignments at laboratory, written test in the last week of the semester or in the exam session
	Final result - formula	A partial grade is established based on the lab results and the written test results. Then the students are classified based on their partial grade, the final grade being assigned accordingly to ECTS grading system– European Credit Transfer System and Diploma Supplement.

COURSE NAME	**RULE BASED PROGRAMMING**	CODE: CS3209O1

STUDY YEAR	III	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	OP

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	94	5	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. DAN CRISTEA	Computer Science

PREVIOUS COURSES REQUESTED	-

OBJECTIVES	The course is an introduction to a programming paradigm that is suitable for creating expert systems. An expert system (ES) is a program that simulates the experience and the reasoning capacities of a human specialist in a certain field.
GENERAL DESCRIPTION	Internal ES design, types of ESs (forward chaining, backward chaining, and mixed chaining), the RETE algorithm (the foundation for optimizing the ES implementations). CLIPS and PERL languages are introduced. Programming techniques in CLIPS are presented in an interactive manner, accompanied by numerous examples.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	The laboratories are used for practicing the writing of programs in CLIPS and PERL. Part of the time will be allocated to working on a project.
TEACHING METHODS	Power Point presentations and interactive programming sessions

BIBLIOGRAPHY (SELECTION)	Dan Cristea, "Programarea bazată pe reguli", Ed. Academiei, 2002 On-line courses posted at http://thor.info.uaic.ro/~dcristea/teaching.html

EVALUATION	conditions	Minimal request to enter the exam: 30 lab points (out of 42) + 20 project points (out of 100) or 20 points on the partial exam Minimum to pass: 26 lab points + 50 project points + 50 written exam points (out of 100)
	criteria	Labs: 12*{● in the class, ●● solved exercise, ●●● remarked} ➔ max 36; project: 0 – 40; written exam (partial + final): 0 – 40
	evaluation methods	Lab, project, written exam
	final result - formula	(lab+pro+ex)/10, modified by Gauss

COURSE NAME	**BIOINFORMATICS**	CODE: CS3209O2

STUDY YEAR	III	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	OP

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	48	144	5	M	Romanian

COURSE TEACHERS	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	ASSOC. PROF. DR. LIVIU CIORTUZ	Computer Science

PREVIOUS COURSES REQUESTED	-

OBJECTIVES	Understanding the basic technics in the analysis of genetic sequences
GENERAL DESCRIPTION	1. Fundamental notions in molecular biology 2. Hidden Markov Models (HMMs) 3. Alignment of pairs of DNA/protein sequences 4. Alignment of pairs of DNA/protein sequences using pair-HMMs 5. Multiple alignment of DNA/protein sequences 6. Multiple alignment of DNA/protein sequences using profile-HMMs 7. Philogenetics; probabilistic models 8. Probabilistic context free grammars (PCFGs) 9. Analysis of RNA sequences using PCFGs 10. Motif in identification in genetic sequences
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Implementation of some of the presented algorithms and their applications. Presentation of recent papers in the area of bioinformatics.
TEACHING METHODS	Slides on video-projector

BIBLIOGRAPHY (SELECTION)	"Biological Sequence Analysis", Durbin, Eddy, Krogh, Mitchison; Cambridge University Press, 1998 "An Introduction to Bioinformatics Algorithms", Neil Jones & Pavel Pevzner; MIT Press, 2004 "Computational Genomics", Nello Cristianini, Matthew Hahn, Cambridge University Press, 2006.

EVALUATION	conditions	Minimum 1.5 (of 4) points for lab, .minimum 1.5 (of 4) puints at the final exam
	criteria	
	evaluation methods	
	final result - formula	Basis (2 points) + lab work (4 points) + final exam (4 points)

COURSE NAME	**THE MODELLING OF DISTRIBUTED SYSTEMS USING PETRI NETS**	CODE: CS3210O1

STUDY YEAR	III	SEMESTER	2	COURSE STATUS (**C**-COMPULSORY/**OP**-OPTIONAL/**F**-FACULTATIVE)	OP

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS OF INDIVIDUAL ACTIVITY	CREDITS	EVALUATION TYPE (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	94	5	P	Romanian

TAUGHT BY	ACADEMIC AND SCIENTIFIC TITLE, NAME	DEPARTMENT
	ASSIST. DRD. OANA OTILIA PRISECARU	Computer Science

REQUIRED COURSES	-

OBJECTIVES	1. Assimilating the basic notions about Petri nets and their different extensions. 2. Acquiring analysis techniques for Petri nets. 3. Understanding the practical applications of Petri nets in various areas. 4. Achieving the ability to model and analyze real systems by using different types of Petri nets.
GENERAL THEMATICS	1. P/T Petri nets. Properties and analysis methods for P/T nets. Subclasses of P/T nets. 2. Applications of P/T nets (modeling communication protocols, distributed algorithms). 3. Worflow theory: workflow nets, properties and extensions of workflow nets. Petri nets-based workflow management systems. 4. Coloured Petri nets: properties and analysis methods. 5. Applications of coloured Petri nets in industry. 6. Timed Petri nets. 7. Applications of timed Petri nets.
SEMINARY / LABORATORY THEMATICS	P/T Petri nets. The modelling and analysis of real systems using P/T nets. The use of simulation and verification tools for P/T nets. Workflow nets. The use of specific tools for workflow nets. The use of a Petri nets-based workflow management system. Coloured Petri nets. System modelling and analysis using coloured Petri nets. Case studies using CPNTools. Timed Petri nets. Other extensions of Petri nets.
TEACHING METHODS	Exposure using course notes (slides, available from the begining of each course) presented with a video-projector, demos on computer. Exposure (video-projector for the course), debate, exercises, problems, case studies (laboratory).

BIBLIOGRAPHY	1. T. Jucan, F.L. Tiplea: Retele Petri. Teorie si Practica. Romanian Academy Press, Bucuresti, 1999. 2. T. Murata. *Petri nets: Properties, analysis and applications*. Proc. of the IEEE 77(4), pp. 541-580, 1989. 3. W. Reisig. *Elements of Distributed Algorithms. Modeling and Analysis with Petri Nets*, Springer-Verlag, 1998. 4. K. Jensen. *Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use*. Vol. 1, Basic Concepts. Monographs in Theoretical Computer Science, Springer-Verlag, 2nd corrected printing 1997. ISBN:3-540-60943-1. 5. W.M.P. van der Aalst and K.M. van Hee. *Workflow Management: Models, Methods, and Systems*. MIT press, Cambridge, MA, 2004. 6. Wil M. P. van der Aalst: *Interval Timed Coloured Petri Nets and their Analysis*. Application and Theory of Petri Nets 1993: 453-472.

EVALUATION	conditions	At least 20 points for the seminary and laboratory activity (LSA). At least 20 point at the written test (TS).
	criterias	A maximum of 100 points can be accumulated.
	modes	Seminary and laboratory activity (exercises, report): 50 points. Written test (TS) in the 14th week: 50 points.
	formula	The final mark is computed by summing the scores (LSA + TS) and then applying Gauss distribution.

COURSE NAME	**ADVANCED OPERATING SYSTEMS**	CODE: CS3211O1

STUDY YEAR	III	SEMESTER	2	COURSE STATUS (**C**-COMPULSORY/**OP**-OPTIONAL/**F**-FACULTATIVE)	OP

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS OF INDIVIDUAL ACTIVITY	CREDITS	EVALUATION TYPE (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	0	2	0	48	102	5	M	Romanian

TAUGHT BY	ACADEMIC AND SCIENTIFIC TITLE, NAME	DEPARTMENT
	LECT. DR. CRSTIAN VIDRAŞCU	Computer Science

REQUIRED COURSES	Computer Architecture, Operating Systems, Programming

OBJECTIVES	This course offers a thoroughgoing study of the basic mechanisms of operating systems and focuses on advanced topics about the kernel modules of operating systems. The operating systems concepts are explained using the Windows XP and Windows Server 2003 operating system family, based on the curriculum developed by the Microsoft Windows Academic Program, structured according with ACM/IEEE Operating System Body of Knowledge.
GENERAL THEMATICS	1. Overview of operating systems 2. Review of basic concepts about operating systems design, concurrency, scheduling, memory management 3. Device management. I/O system. File systems 4. Real-time and embedded systems 5. Fault tolerance 6. System performance evaluation and troubleshooting 7. Scripting 8. Comparing the Linux and Windows kernels 9. Windows and Unix interoperability
SEMINARY / LABORATORY THEMATICS	Lab works related to the course' general thematics. The students will do lab experiments and assignments through which they will study operating systems concepts related to Windows XP and Windows Server 2003 operating system family, using the resources from Windows Academic Program.
TEACHING METHODS	Exposure using course notes (slides, available from the begining of each course) presented with a video-projector, demos on computer.

BIBLIOGRAPHY	8. 1. Mark Russinovich and David Solomon, Windows Internals, 4th edition, Microsoft Press, 2005. 2. Windows Academic Program: http://www.microsoft.com/resources/sharedsource/windowsacademic/default.mspx

EVALUATION	conditions	The presence at the laboratory activities and at the written thesis.
	criterias	Minimal score for graduation is 30p
	modes	Evaluation during the semester (lab works) and a final written thesis.
	formula	Final Score = Lab * 0.5 + WrittenThesis * 0.5. The final mark is obtained from the final score through classification based on the ECTS – European Credit Transfer System and Diploma Supplement.

COURSE NAME	**EMBEDDED SYSTEMS**	CODE: CS3211O2

STUDY YEAR	III	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	OP

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	94	5	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	LECT. DR. VLAD RĂDULESCU	Computer Science

PREVIOUS COURSES REQUESTED	Computer Architecture and Operating Systems, Hardware Practice, Operating Systems

OBJECTIVES	Getting acquainted with the concept of embedded system. Learning the hardware design requirements. Learning the ways of writing software for embedded systems.
GENERAL DESCRIPTION	Hardware and software design of embedded systems. Simple and complex hardware automata. Hardware description languages. Real-time systems. Restrictions. Microprocessors and microcontrollers. I/O systems. Interrupts. Time measurements. Signal processing. Analog and digital signals. Conversions. Data acquisition.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Simple and complex hardware automata. The Verilog hardware description language. Buses. Arbitration tehniques. Programming the microcontrollers.
TEACHING METHODS	Exposition, problem-solving, case studies, exercises.

BIBLIOGRAPHY (SELECTION)	
	H. Kopetz, *Real-time Systems*, Kluwer Academic Publishers, 1997. C. M. Krishna, K. Shin, *Real-time Systems*, Mc-Graw Hill, 1997. D. Lewis, *Fundamentals of Embedded Software*, Prentice-Hall, 2001.

EVALUATION		
	conditions	The presence at the laboratory activities.
	criteria	At least 5 points on each test; 3 or 4 points are accepted for at most one test.
	evaluation methods	Two written tests, the first (T1) during the 7th week, the second (T2) during the examination session.
	final result - formula	Final result = (T1 + T2) / 2, on which Gaussian distribution is applied.

COURSE NAME	**ANVANCED SOFTWARE ENGINEERING TECHNICS**	CODE: MCG1101

STUDY YEAR	I	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	124	8	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. BAZIL PÂRV	Computer Science

PREVIOUS COURSES REQUESTED	

OBJECTIVES	The development of a good understanding of the software engineering field. Students will get to learn concepts used in designing complex systems, modern architectures and new approaches in modelling competitive software.
GENERAL DESCRIPTION	0 SWEBOK: THE ROLE OF SOFTWARE ENGINEERING WWW.SWEBOK.ORG KNOWLEDGE AREAS RELATED COURSES STEVE MCCONNELL (WWW.CONSTRUX.COM) PRAGMATIC PROGRAMMER 1 OO DESIGN - CLASSES: GRASP (LARMAN), RESPONSIBILITY-DRIVEN DESIGN - MID-LEVEL: GOF (OVERVIEW) - HIGH-LEVEL: ARCHITECTURAL STYLES (PATTERNS), SOA - OO DESIGN PRINCIPLES 2 SYSTEM DEVELOPMENT AND MAINTENANCE THE CHARACTERISTICS OF A GOOD ARCHITECTURE (RCM) AGILE MODEL-DRIVEN DEVELOPMENT ENTERPRISE APPLICATION ARCHITECTURE PATTERNS DOMAIN-DRIVEN DESIGN: CONCEPT AND PATTERNS TEST-DRIVEN DEVELOPMENT REFACTORING: CODE (R IN THE SMALL), ARCHITECTURE (R IN THE LARGE) 3 MODELING BUSINESS MODELING: BPMN UML BEHAVIORAL MODELS: STATE MACHINES, ACTIVITIES WORKFLOW PATTERNS MODEL-DRIVEN DEVELOPMENT, MODEL-DRIVEN ARCHITECTURE DOMAIN SPECIFIC LANGUAGES (DSL), FRAMEWORKS: ECLIPSE MODELING FRAMEWORK, OPEN ARCHITECTURE WARE (OAW)
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Applying software engineering best practices. Refactoring, improving the design of existing code. Automated testing Advanced design patterns
TEACHING METHODS	Course slide presentations. Lecture notes and tutorials available electronically.

BIBLIOGRAPHY (SELECTION)	Guide to the Software Engineering Body of Knowledge http://www.swebok.org/ The Pragmatic Programmer: From Journeyman to Master by Andrew Hunt and David Thomas Refactoring Home Page, http://www.refactoring.com/ Martin Fowler homepage, http://martinfowler.com/

EVALUATION		
	conditions	Participation in practical works. Accumulation of 50 points, according to the final grade formula.
	criteria	Written exam (T, max. 40), article (R, max. 50), practical project (P, max. 60)
	evaluation methods	Written exam at the end of the semester. The article and the project will be evaluated as part of the laboratory works.
	final result - formula	F = P + R + T On the final grade a Gauss like distribution will be applied, according to the current regulations.

COURSE NAME	**OPERATIONS RESEARCH**	CODE: MCG1102

STUDY YEAR	I	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	124	8	P	Romanian

COURSE TEACHERS	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	ASSOC.PROF. DR. MARIANA RODICA BRÂNZEI	Computer Science

PREVIOUS COURSES REQUESTED	Mathematics Probability Theory and Statistics

OBJECTIVES	Operations Research (OR) is the application of scientific techniques and methodology to decision-making problems. The purpose of the course is to provide a sound understanding of several quantitative models and methods which are frequently and successfully applied in managerial decision making, and more generally, to demonstrate the possibilities and limitations of the quantitative approach in the analysis of decision situations.
GENERAL DESCRIPTION	- Overview of the OR Modelling Approach; - Linear Programming (LP); Graphical LP solutions; The Simplex Method; - Duality Theory and Sensitivity Analysis; - Integer Programming; - Game Theory and Decision Analysis, - Queueing Theory; - Inventory Theory.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	The OR laboratory is application oriented and is mainly build on the case method. The case method creates a classroom in which students succeed to understand theory and exercise the skills of leadership and team work in the face of real problems. Several types of managerial decision problems are described which can be adequately represented by quantitative models and can be solved by mathematical methods. The emphasis is on model formulation. Computer solutions by using spreadsheet programs (like Excel) and computer interpretations in the context of the decision situations are also demonstrated.
TEACHING METHODS	Lectures using overhead projector and blackboard.

BIBLIOGRAPHY (SELECTION)	1. Hamdy A. Taha, Operations Research: An Introduction, 8/E, Pearson, 2008; 2. Frederick S. Hillier and Gerald J. Lieberman, Operations Research, 8/e, Mc Graw Hill, 2005; 3. Rodica Branzei, Dinko Dimitrov and Stef Tijs, Models in Cooperative Game Theory, Springer, 2008.

EVALUATION	conditions	ME (meadterm evaluation), FE (final evaluation), W (Laboratory works)
	criteria	$ME \geq 6$, $FE \geq 4$, $W \in \{0, 1, 2\}$
	evaluation methods	ME (written test (90 minutes) covers 1-6 weeks; FE (written test (90 minutes) covers weeks 8-13)
	final result - formula	ME + FE + W

COURSE NAME	**JAVA TECHNOLOGIES**	CODE: MCG1103

STUDY YEAR	I	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	154	8	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	LECT. DR. CRISTIAN FRĂSINARU	Computer Science

PREVIOUS COURSES REQUESTED	Object-oriented programming, Advanced Techniques of Programming, Web Technologies, Networking

OBJECTIVES	Java Enterprise Edition (Java EE) , Server-Side Programming
GENERAL DESCRIPTION	Introduction in Java Enterprise Edition (Java EE) Web Components: Servlets, Filters, Java Server Pages (JSP), Custom Tag Libraries (CTL) Template Engines: Velocity, Free Marker MVC frameworks: Struts, Java Server Faces Java Naming and Directory Interface (JNDI). Java Message Service (JMS). Object-Relational Mappings: Hibernate, Java Persistence API (JPA) Object-Oriented Databases DB4O. Aspect Oriented Programming (AOP): AspectJ. Service Oriented Architectures (SOA). Enterprise Java Beans (EJB)
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Problems concerning each individual course.
TEACHING METHODS	Videoprojection

BIBLIOGRAPHY (SELECTION)	Cristian Frasinaru, *Curs practic de Java*, Matrix Rom Bucuresti (2005), ISBN 973-685-856-1 Jayson Falkner, Kevin Jones, *Servlets and Java Server Pages*, Ed Roman, Scott Ambler, Tyler Jewel, *Mastering Enterprise JavaBeans* http://java.sun.com/javaee

EVALUATION	conditions	Each laboratoy will contain two problems, each counted with 1 point. The exam will contain 20 questions, each counted with 1 point. Supplimentary work mai count additional points.
	criteria	To enter exam, each student must have 8 points. To pass exam, each student must have 5 points.
	evaluation methods	Problem presentation (during semester) and Exam (in session)
	final result - formula	Gauss curve on the total number of points 5%=10, 10%=9, 20%=8, 30%=7, 25%=6, 10%=5

COURSE NAME	**ADVANCED ARTIFICIAL INTELLIGENCE (MACHINE LEARNING)**	CODE: MCG1205

STUDY YEAR	I	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	184	8	M	Romanian

COURSE TEACHERS	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	ASSOC. PROF. DR. LIVIU CIORTUZ	Computer Science

PREVIOUS COURSES REQUESTED	-

OBJECTIVES	Understanding the conception and practical usage of the most important algorithms for classification and clustering
GENERAL DESCRIPTION	1. Concept Learning and General-to-Specific Ordering 2. Decision Trees 3. Instance-based Learning 4. Retele neuronale 5. Bayesian Learning 6. Support Vector Machines 7. Clustering Algorithms 8. Boosting, bagging, random forests, voting, co-training 9. Evaluating Hypotheses 10. Computational Learning Theory 11. Reinforcement Learning
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Presentation of recent papers in the area of machine learning: String kernels and applications Pattern matching on strings Grammar learning
TEACHING METHODS	Slides on video-projector

BIBLIOGRAPHY (SELECTION)	„Machine Learning”, Tom Mitchell; McGraw-Hill, 1997 „Data Mining: Practical ML Tools and Techniques with Java Implementations", Witten, Frank; Morgan Kaufmann Publishers, 2000 „The Elements of Statistical Learning", Friedman, Hastie, Tibshirani, 2001 „Pattern Matching and Machine Learning”, Ch. Bishop, 2006

EVALUATION		
	conditions	Minimum 1.5 (of 4) points for lab, .minimum 1.5 (of 4) puints at the final exam
	criteria	
	evaluation methods	
	final result - formula	Basis (2 points) + lab work (4 points) + final exam (4 points)

COURSE NAME	**WEB APPLICATION DEVELOPMENT**	CODE: MCG2101

STUDY YEAR	II	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	124	8	P	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	ASSOC. PROF. DR. SABIN-CORNELIU BURAGA	Computer Science

PREVIOUS COURSES REQUESTED	Web Technologies, Software Engineering

OBJECTIVES	Giving a general view regarding knowledge modeling in the context of evolution towards semantic Web. Students will achieve understanding about the development of knowledge management Web applications on the basis of the present Web technologies: metadata, microformats, taxonomies, thesauri, and ontologies.
GENERAL DESCRIPTION	Important concepts. Terminology. The evolution of the World Wide Web space. Revisiting Web architecture. Web application architecture. Aspects regarding Web performance. Knowledge modeling in the context of social and semantic Web. Architecture of the semantic Web-based applications. Specification of metadata and relations between resources. RDF (*Resource Description Framework*). Characteristics. Conceptual model. Alternative syntaxes. SPARQL. Examples and applications. Ontologies. Definitions and characteristics. Types. Specification techniques. Taxonomies. RDF Schema. Examples. Thesauri. SKOS (*Simple Knowledge Organizational System*). Other conceptual models. OWL (*Web Ontology Language*). Examples. Formal specification of ontologies. Introduction to description logic. Ontological engineering. Methodologies and case studies. Specification of rules. Support for automatic reasoning. Examples. Software agents. General overview. Multi-agent systems. Case studies. Semantic Web services. Advanced aspects concerning SOA (*Service Oriented Architecture*). Ontologies for Web services (OWL-S. WSMO). Semantic *mash-ups*. Semantic grid services. Grid application architecture. Using semantic Web services and agents in the context of grid computing.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Conceptual modeling of XML data. Native XML databases. XQuery. Attaching metadata to Web resources. RDF. Microformats. RDFa. SPARQL. Specification of taxonomies and thesauri. Expressing ontologies via OWL. Case studies.
TEACHING METHODS	Interactive presentations. Direct interaction. Online access to educational resources via the Website course.

BIBLIOGRAPHY (SELECTION)	8. D. Allemang, J. Hendler, *Semantic Web for the Working Ontologist*, Morgan Kaufmann, 2008. 9. H. P. Alesso, C. F. Smith, *Thinking on the Web*, John Wiley & Sons, 2006. 10. G. Antoniou, F. van Harmelen, *A Semantic Web Primer* (2nd Edition), MIT Press, 2008. 11. S. Buraga, *Tehnologii XML* (in Romanian), Polirom, Iaşi, 2006. 12. S. Buraga, *Semantic Web* (in Romanian), Matrix Rom, 2004. 13. M. Daconta, L. Obrst, K. Smith, *The Semantic Web*, John Wiley & Sons, 2003. 14. N. Josuttis, *SOA in Practice*, O'Reilly, 2007. 15. R. Yee, *Pro Web 2.0 Mashups: Remixing Data and Web Services*, Apress, 2008. 16. * * *, *Semantic Web*: *http://www.semanticweb.org/* * * *, *World Wide Web Consortium*: *http://www.w3.org/TR/*

EVALUATION		
	conditions	1 project (P), 1 test during semester (T)
	criteria	project P>5, test T>5
	evaluation methods	1 project (P), 1 test during semester (T)
	final result - formula	0.5 * P + 0.4 * T + 1

COURSE NAME	**PROJECT MANAGEMENT**	CODE: MCG2205

STUDY YEAR	II	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	184	8	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. GABRIELA MEŞNIŢĂ	Faculty of Economics and Business Management

PREVIOUS COURSES REQUESTED	Management; Software Engineering

OBJECTIVES	To create the project management feels To develop a specific language To get abilities for project design To achieve different information and knowledge concerning the project management, assessement and evaluation To use the project management methods and tools
GENERAL DESCRIPTION	Conceptual framework of project management Team work and team management of project management Practical approach of project plan Cost control and project budget Proposal project evaluation Project monitoring and control Project management and information technology
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Team work focused on design a project following: identification the right phases of project development; reference terms and requirements of different types of project; project initiation; project planning; Project assessement; use specific software (Microsoft Project).
TEACHING METHODS	Interactive courses, practical home work, team work

BIBLIOGRAPHY (SELECTION)	Kerzner, H. – Project Management, John Wiley & Sons, Inc., New York, 2001. Kezsbom, D.S., Edward, K.A. – The New Dynamic Project Management, Second Edition, John Wiley & Dons, Inc., New York, 2001. Lientz, B.P., Rea, K.P. – Guide to Successful Project Management, Harcourt Brace Professional Publishing, San Diego, CA, USA, 1999. Oprea, D., Managementul proiectelor. Teorie şi cazuri practice, Sedcom Libris, 2001. Internet addresses: www.wst.com, www.pmi.org, www.gantthead.com, www.allpm.com, www.projectmagazine.com, www.cordis.lu, www.finatare.ro, www.infoeuropa.ro

EVALUATION		
	conditions	Lab project; Home work, Partial evaluation at cours and lab, Writing exam
	criteria	Each condition must be evaluated at least with 5
	evaluation methods	Project presentation (40%); Home work (10%); Partial evaluation (10%); Writing exam (40%)
	final result - formula	Project x 0,4 + Home work x 0,1 + Partial evaluation x 0,1 + Exam x 0,4

COURSE NAME	**PARALLEL ALGORITHMS AND PARALLEL PROGRAMMING**	CODE: MSD1206

STUDY YEAR	I	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	184	8	E	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	LECT. DR. LUCIAN GHIRVU	Computer Science

PREVIOUS COURSES REQUESTED	BSc in Computer Science

OBJECTIVES	PARALLEL ALGORITHMS : this course is an introduction in the design and the analysis of parallel algorithms and also presents the main models of parallel computation. PARALLEL PROGRAMMING : there are presented its main concepts and applications (using the libraries such as Pthreads, MPI, OpenMP).
GENERAL DESCRIPTION	PARALLEL ALGORITHMS : parallel computer/algorithm, the analysis of parallel algorithms, parallel computation models (PRAM, interconnection networks, combinational circuits), parallel prefix computation, using the „Divide and conquer" paradigm in the design of parallel algorithms (for searching, merging, and selection), using of pointer-based data structures in parallel algorithms. PARALLEL PROGRAMMING : using of the libraries such as Pthreads, MPI, OpenMP.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Problems (projects). The implementation of some parallel algorithms and solving problems by using the libraries C Pthreads, MPI, and OpenMP. Case studies.
TEACHING METHODS	Exposition (lectures are given using blackboard presentations).

BIBLIOGRAPHY (SELECTION)	
	C.Croitoru. Introducere in proiectarea algoritmilor paraleli. Ed.Matrix Rom, 2002. R.W. Hockney, C.R. Jesshope. Calculatoare paralele. Ed. Tehnică, 1991. D.Gâlea, O.Brudaru. Introducere în calculul sistolic. Ed. Academiei Române, 1994. I.Chiorean. Calculul paralel, fundamente. Ed. Albastră, 1995. G. Andrews. Foundations of Multithreaded, Parallel, and Distributed Programming. Addison Wesley 2000. Papers from journals and conferences.

EVALUATION		
	Conditions	No absence at laboratory classes.
	Criteria	Activity at laboratory classes (pertinent questions regarding the assignments, presentation of several stages of solving the assignment), strong lectures attendance (in case of low levels of attendance, the attendance of a particular student could be recompensed by granting bonuses to his/her final grade).
	Evaluation methods	Programming assignments at laboratory, written test in the last week of the semester or in the exam session, scientific paper presentations.
	Final result - formula	A partial grade is established based on the results of the above evaluation methods. Then the students are classified based on their partial grade, the final grade being assigned accordingly to ECTS grading system– European Credit Transfer System and Diploma Supplement.

COURSE NAME	**SECURITY PROTOCOLS**	CODE: MSD2207

STUDY YEAR	I	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	184	8	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	DR. CONSTANTIN ENEA	Computer Science

PREVIOUS COURSES REQUESTED	Familiarity with basic cryptographic concepts as treated, for example, in the core course "Information Security", is helpful, but can in principle also be acquired in parallel to attending the course.

OBJECTIVES	The objectives of the course are to provide students with the practice of cryptographic protocols (efficient implementations, vulnerabilities, etc.) and to provide a background against which the student will be able to develop new protocols and applications.
GENERAL DESCRIPTION	Topics include: • basics on security (cryptographic) protocols • formalisms: MSR, strand spaces • undecidability of secrets • bounded protocols • taged protocols • recursive protocols • verification techniques: BAN,inductive method, strand spaces,based on I/O-automata
DESCRIPTION OF SEMINARY / LABORATORY WORK	All seminars will be oriented on the topic discussed during the courses. Students will be asked to prepare a research project.
TEACHING METHODS	On-line and blackboard presentation.

BIBLIOGRAPHY (SELECTION)	• F.L.Tiplea: Algebraic Foundations of Computer Science, Polirom, 2006. • F.L.Tiplea: *Introduction to Cryptography* (in preparation) - chapters of the book will be available to students. • Research articles.

EVALUATION	conditions	
	criteria	
	evaluation methods	One project and a final exam.
	final result - formula	50% from the project and 50% from the final exam.

COURSE NAME	**COMBINATORIAL OPTIMIZATION**	MOC1206

STUDY YEAR	I	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	2	-	-	56	124	8	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. CORNELIUS CROITORU	Computer Science

PREVIOUS COURSES REQUESTED	Design and Analysis of Algorithms, Graph Algorithms, Operational Research

OBJECTIVES	This course is an advanced study in the design and analysis of algorithms related to combinatorial optimization using SAT, a classical problem in theoretical computer science, which was widely investigated and has brought forward a rich body of methods and tools, both in theory and practice. To provide basis for independent research on the subject.
GENERAL DESCRIPTION	The course concentrates on the combinatorial and algorithmic aspects of the propositional logic Satisfiability Problem. The fundamental challenge is understanding of Combinatorial Search Spaces.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Each seminary debates recent reasearch papers in order to deepen the subjects introduced in the course. All these papers are posted at the begining of the semester such that interested students could try to study in advance.
TEACHING METHODS	Video presentations of the slides (containing the course notes) available in pdf format at the begining of each class.

BIBLIOGRAPHY (SELECTION)	
	- A repository of about 200 papers covering the material of the course will be distributed. - Emo Welzl : Course on Satisfiability of Boolean Formulas – Combinatorics and Algorithms (http://www.ti.inf.ethz.ch/ew/courses/SAT08/) - http://www.satlive.org/SATBIB/

EVALUATION		
	conditions	
	criteria	A student will be considered to have passed the exam if (s)he obtains at least 50 points.
	evaluation methods	-Seminary activity (attendance, work quality): 0-20 points. - Homeworks : 0-40 points. - Written Final test : 0-40 points.
	final result - formula	The final grade (if the total number of points is at least 50) is given by applying the ECTS rules.

COURSE NAME	**MULTIMEDIA TECHNOLOGIES IN AUTOMOTIVE**	CODE: MISS1206

STUDY YEAR	MASTER 1	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	0	2	0	56	94	8	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	EXPERT OF THE COMPANY CONTINENTAL AUTOMOTIVE ROMANIA	Computer Science

PREVIOUS COURSES REQUESTED	Advanced Software Engineering Technics Algorithms and Programming Object Oriented Programming Software Engineering

OBJECTIVES	Assimilating the basic information needed for creating a multimedia application in the automotive environment. Fundamental knowledge of the domain, workflow, procedures and quality standard required in the field.
GENERAL DESCRIPTION	General presentation of the multimedia automotive systems. The V-Cycle. Project planning. Requirements creation and analysis, functional and technical specifications. System architecture, software architecture, detailed software design. Human Machine Interface. Navigation. Entertainment. Connectivity. Speech. Testing techniques. System Integration. Design for testability. Automated testing. Validation. Project closure. Archiving, Lessons learned. Specific tools.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Presentation of hardware currently used in the industry and of an actual system developed in Iasi (VW RNS). Familiarization with the devices used for laboratory classes, technical capabilities. Creating the specifications for an actual application. Creating the architecture. Software design for the application. HMI functionality implementation. Navigation functionality implementation. Entertainment functionality implementation. Connectivity functionality implementation. Speech functionality implementation. Test cases for the application. Performance measurements. Automated testing. Archiving the project, lessons learned workshop. Applications with the specific tools. "Find bugs" contest.
TEACHING METHODS	Presentations for the courses. Workshops with FCS and CONTINENTAL AUTOMOTIVE ROMANIA provided software and hardware during the laboratories. Questions and answers with expert guests.

BIBLIOGRAPHY (SELECTION)	Bjarne Stroustrup: *The C++ Programming Language*, *Adisson-Wesley*, 3rd edition, 1997 Boris Beizer: *Software Testing Techniques* Erich Gamma, Richard Helm , Ralph Johnson, John Vlissides: *Design Patterns: Elements of Reusable Object-Oriented Software* T. Vaughan: *Multimedia: Making it Work*, 6th Edition, *McGraw-Hill Osborne Media*, 2003.

EVALUATION	conditions	Laboratory involvement and results (L) – 60%. Written examination for the courses (WE1, WE2) – 40%.
	criteria	More then 50p in total (maximum of 100p). L > 30p, WE1, WE2 > 10p.
	evaluation methods	L – the maximum 60p will be divided between the 14 laboratories. Evaluations will be made for each laboratory, taking into account the overall involvement (bonuses) and the results relative to the targets. Specific criteria will be provided before each laboratory. There will be no homework assignments. WEs – the written tests will verify the level of understanding and assimilation for the information presented during the courses.
	final result - formula	Final grade = (L + WE1 + WE2) / 10

COURSE NAME	**SPECIAL CHAPTERS IN HUMAN-COMPUTER INTERACTION**	CODE: MISS1207

STUDY YEAR	I	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	124	8	P	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	ASSOC. PROF. DR. SABIN-CORNELIU BURAGA	Computer Science (Master)

PREVIOUS COURSES REQUESTED	Software Engineering, Computer Graphics, Web Technologies

OBJECTIVES	To provide a comprehensible vision of the user interface design aspects and the user-computer interaction types. The students will be able to design ergonomic and attractive interfaces for specific types of users, applications, platforms, and devices, including Web and mobile ones.
GENERAL DESCRIPTION	Definitions and terminology. Aspects regarding the human-computer interaction. User interface design: principles, models, and types. Design methodologies. Case studies: game development. Human factor. Usability. Program presentation – at the desktop level. User interaction. Input devices. Graphical controls. Output devices. Components. Affectivity, errors and risks. User education and guidance. Interface identity and evaluation. User testing. Methodologies. High-level specification of interfaces. Web interaction. Web interface design. Methodologies. Case studies. Non-conventional interactions. From mobile interaction to ubiquitous computing. Augmented and virtual reality.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Human-computer interaction patterns. Issues on interaction and user-interface design. Specific user-interface prototyping. Techniques of user interface evaluation. User testing.
TEACHING METHODS	Interactive presentations. Direct interaction. Online access to additional resources via the Website course.

BIBLIOGRAPHY (SELECTION)	11. S. Buraga, *Proiectarea siturilor Web* – ediția a II-a (in Romanian), Polirom, Iaşi, 2005. 12. A. Cooper, R. Reimann, D. Cronin, *About Face* (3rd Edition), Wiley Publishing, 2007. 13. B. J. Fogg, *Persuasive Technology*, Morgan Kaufmann Publishers, 2003. 14. B. Fry, *Visualising Data*, O'Reilly, 2008. 15. E. Law, E. Hvannberg, G. Cockton (Eds.), *Maturing Usability*, Springer, 2008. 16. J. Novak, *Game Development Essentials* (2nd Edition), Thomson, 2008. 17. D. Safer, *Designing for Interaction: Creating Smart Applications and Clever Devices*, Peachpit Press, 2006. 18. A. Sears, J. Jacko (eds.), *The Human–Computer Interaction Handbook* (2nd Edition), Taylor & Francis Group, 2008. 19. * * *, *HCI Design Patterns*: http://www.hcipatterns.org/ 20. * * *, *UsiXML Consortium*: http://www.usixml.org/

EVALUATION		
	conditions	1 project (P), 1 test during semester (T)
	criteria	project P>5, test T>5
	evaluation methods	1 project (P), 1 test during semester (T)
	final result - formula	0.5 * P + 0.4 * T + 1

COURSE NAME	**SOFTWARE SECURITY**	CODE: MISS2103

STUDY YEAR	II	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	184	8	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. GHEORGHE GRIGORAS	Computer Science

PREVIOUS COURSES REQUESTED	No prerequisite required.

OBJECTIVES	The course is an introduction in various programming-based methods for the development of security policies. Students will acquire experience with programming with various Java security-related packages and with access control modules in Linux.
GENERAL DESCRIPTION	The topics covered are: 1. Access control in Java 2. The Java Security and Cryptography extensions 3. SELinux, a Linux module for access control 4. Jif, a package for controlling information flow in Java programs 5. JAAS, the Java Authentication and Authorization System
DESCRIPTION OF SEMINARY / LABORATORY WORK	Seminars and laboratories are grouped around the chapter currently discussed in the course. They aim to illustrate the topics of the chapter mainly by practical applications.
TEACHING METHODS	On-line and blackboard presentation.

BIBLIOGRAPHY (SELECTION)	1. Scott Oaks, *Java Security*, O'Reilly, ISBN 978-0596001575 2. Frank Meyer, David Kaplan, Karl McMillan, *SELinux by Examples*, Prentice Hall PTR, ISBN 978-0131963696 3. Tutorial pages for JAAS at http://java.sun.com/javase/6/docs/technotes/guides/security/jaas/JAASRefGuide.html 4. Tutorial pages for Jif at http://www.cs.cornell.edu/jif/

EVALUATION		
	conditions	
	criteria	
	evaluation methods	7 homeworks and a final exam.
	final result - formula	50% from the homeworks and 50% from the final exam.

COURSE DESCRIPTION

COURSE NAME	**QUALITY SYSTEMS SOFTWARE**	CODE: MISS2207

STUDY YEAR	II	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2		2	-	56	184	8	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. DOREL LUCANU	Computer Science

PREVIOUS COURSES REQUESTED	Engineering software systems (programming Engineering)

OBJECTIVES	Understanding the main elements that define the quality of software. Familiarization with the testing methods and analysis software.
GENERAL DESCRIPTION	What is the quality of software? Quality Engineering Quality Testing software systems (concepts, activities, techniques, case studies) beyond quality assurance test (code inspection, formal verification, fault tolerance to) improve the quality of quantification (monitoring, measuring, quality models, analysis and classification of defects) systems analysis software.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Working for individual projects that reflect the different activities to assure quality. Presentation of case studies. Reports.
TEACHING METHODS	Lectures on the system using interactive videoproiectorul, practical work

BIBLIOGRAPHY (SELECTION)	Robert V. Binder Testing Object-Oriented Systems: Models, Patterns, and Tools. Addison-Wesley Professional, 1999 Jeff Tian. Software Quality Engineering: Testing, Quality Assurance, and Quantifiable Improvement. John Wiley and Sons, Inc., and IEEE Computer Society Press, 2005

EVALUATION		
	conditions	Laboratory activity. Themes. Written exam
	criteria	Each condition must be fulfilled to obtain at least a grade 5
	evaluation methods	Laboratory work (40%); Themes 20%, written examination (40%), presentation of reports is a bonus.
	final result - formula	Activity laboratory x 0.4 + theme x 0.2 + Exam x 0.4 + Reports

COURSE NAME	**INTRODUCING NATURAL LANGUAGE PROCESSING**	CODE: ML1206

STUDY YEAR	I	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	124	8	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. DAN CRISTEA, PROF. DR. DAN TUFIS	Computer Science

PREVIOUS COURSES REQUESTED	-

OBJECTIVES	The course of human language technologies to master students, deepening the knowledge taught in the course of Artificial Intelligence in the third year, in the direction of natural language processing and computational linguistics. Among the knowledge that circumscriu this vast area at the intersection between linguistics and computer science, to discuss ongoing problems is the understanding of semantic content of expression in natural language and speech analysis, the ultimate goal being the construction of models for automatic interpretation of texts.
GENERAL DESCRIPTION	Classical theories of discourse (alert states theory, theory of rhetorical structure theory centers), problems of cohesion text (anaphoric and its resolution), stringers theory, incremental parsing text, summarizing text, ontologies language applications.
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Closely follows the concepts taught in class. Theme laboratories are posted at http://profs.info.uaic.ro/~ipistol/tlu0708
TEACHING METHODS	Power Point presentations

BIBLIOGRAPHY (SELECTION)	Cristea & Dima, 2001: "An Integrating Framework for Anaphora Resolution", Information Science and Technology, Romanian Academy Publishing House, Bucharest, vol 4, no. 3 Grosz, B.; Joshi, A.K. and Weinstein, S: "Centering: A framework for modeling the local coherence of discourse, Computational Linguistics, 21 (2), June, 1995. Mann, W. Thompson, S. rhetorical Structure Theory, 1987. Cristea, D.; Webber, B.L. (1997): Expectations in Incremental Discourse Processing. In Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics, Madrid, 8 p. Cristea, D., Ide, N., Romary, L. (1998): Veins Theory. An Approach to Global Cohesion and Coherence. In Proceedings of Coling / ACL'98, Montreal, 5 p. Cristea, D., Ide, N., Mark, D; Tablan, V. (1999): Discourse Structure and Co-Reference: An Empirical Study. In Proceedings of the Workshop on the Relation Between Discourse Structure and Reference, ACL'99, University of Maryland, p. 8 Courses posted on the web at http://thor.info.uaic.ro/~dcristea/teaching.html

EVALUATION	conditions	The minimum requirement for entry into the examination: laboratory 30 points (from 36) + 50 points project (of 100) The minimum requirement for passage: 26 points 50 points laboratory + + Project 50 points sentence (of 100)
	criteria	Laboratories: 12 * {● present ● ● solved year, ● ● ● noted } ➔ max 36; Project: 0 - 100; written exam: 0 - 100
	evaluation methods	Laboratory project, written exam
	final result - formula	Final Note: (100/36 * 1.1 * 1.2 * lab pro + ex) / 30, corrected by Gauss curve

COURSE NAME	**NATURAL LANGUAGE PROCESSING BY STATISTICAL METHODS**	CODE: ML1207

STUDY YEAR	I	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C
CLASS	MASTER OF COMPUTATIONAL LINGUISTICS, 2008 - 2010				

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2		2		56	184	8	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. DAN TUFIŞ	Computer Science

PREVIOUS COURSES REQUESTED	Probability and Statistics – CS 1210 Formal Languages, Automata and Compilers – CS 2103 Artificial Intelligence – CS 3103

OBJECTIVES	The introduction, development and deepening of some elements of linguistic and statistical corpora processing Deepening the modalities of representation and manipulation of linguistic knowledge Development of modeling capacities for building domain applications
GENERAL DESCRIPTION	1. History, terminology, objectives, computational applications of the NLP domain 2. Knowledge representation needed in NLP; stages of NLP, general structure of an automatic NLP systems NLP paradigms: symbolic / statistical approach 3. Linguistic resources and their standardizing ; notions of statistics 4. Zipf's laws in the analysis of large collections of texts 5. Entropy and entropic models; collocations, coligations; mutual information, statistical tests 6. Statistical language models; speech recognition; morpho-lexical disambiguation 7. Hidden Markov models of order *n* 8. Training corpora for morpho-lexical disambiguation; hierarchical morpho-lexical disambiguation and combined language models; applications 9. Automatic word-sense disambiguation 10. Probabilistic syntactic analysis; algorithms of syntactic analysis; stochastic grammars; training corpora (tree-banks)
DESCRIPTION OF SEMINARY / LABORATORY WORKS	Development, annotation and mark-up standards for corpora Statistical processings and tests on corpora Mono- and multi-lingual annotation of corpora for the lexical, morpho-syntactic, semantic, and discourse level
TEACHING METHODS	Interactive presentations combined with free discussions and debates

BIBLIOGRAPHY (SELECTION)	Charniak, E. *Statistical Language Learning*, MIT Press, 1993 Manning, C. Shutze, H. *Fundamentals of Statistical Natural Language Processing*, MIT Press, 1999 Tufiş, D. Algorithms and Data Design Issues for Basic NLP Tools. In Sergei Nirenburg and Oleg Kapanatze (eds.) *Advances in Language Engineering for Low- and Middle-Density Languages*. NATO-ASI, September 2008. 48 p. IOS Press. Tufiş, D., Andersen, P.(eds). *Recent Advances in Romanian Language Technology*, Editura Academiei, 1997 Tufis, D. Filip, F (eds). *Limba Română în Societatea Informaţională - Societatea Cunoaşterii*, Editura Academiei, 2002. Tufiş, D. Tiered Tagging and Combined Classifiers. In F. Jelinek, E. Nöth (eds) *Text, Speech and Dialogue, Lecture Notes in Artificial Intelligence* 1692, Springer, 1999. S. Armstrong, et al. (eds). *Natural Language Processing Using Very Large Corpora*, Kluwer, 1999

EVALUATION		
	conditions	Achieving homeworks and the semestrial project Participation in the laboratory hours
	criteria	Minimal achieving the homework and the semestrial project Active participation in the laboratory classes
	evaluation methods	Final written exam (60%) Homeworks + project (40%)
	final result - formula	NF = 0,6*E + 0,4*R + B Where E = **E**xam grade, R = grade for homeworks and project, B – bonus for outstanding activity

COURSE NAME	**APPLIED CRYPTOGRAPHY**	CODE: MSI1104

STUDY YEAR	I	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
-	-	4	-	56	184	8	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. FERUCIO LAURENȚIU ȚIPLEA LECT. DR.SORIN IFTENE	Computer Science

PREVIOUS COURSES REQUESTED	Familiarity with basic cryptographic concepts as treated, for example, in the core course "Information Security", is helpful, but can in principle also be acquired in parallel to attending the course.

OBJECTIVES	The objectives of the course are to provide students with the practice of cryptographic algorithms and protocols (efficient implementations, vulnerabilities, etc.) and to provide a background against which the student will be able to assess existing applications of cryptography and develop new protocols and applications that employ cryptography.
GENERAL DESCRIPTION	Topics include: • one-way functions, pseudo-randomness, hash functions • symmetric encryption and authentication systems • public-key encryption systems and PKI • digital signature schemes • cryptographyc protocols • cryptanalytic techniques • applications and case studies
DESCRIPTION OF SEMINARY / LABORATORY WORK	During the course, students will be required to complete a project in applied cryptography. These projects must be completed in small groups (no more than 3 students) and each group will be required to present their progress to the other students twice during the course.
TEACHING METHODS	On-line and blackboard presentation.

BIBLIOGRAPHY (SELECTION)	• E. Kranakis. *Primality and Cryptography*, John Wiley and Sons, 1987. • A.J. Menezes, P.C. van Oorschot, S.A. Vanstone. *Handbook of Applied Cryptography*, CRC Press, fifth printing, 2001. • S. Vaudenay: A Classical Introduction to Cryptography, Springer, 2006. • F.L.Tiplea: Algebraic Foundations of Computer Science, Polirom, 2006. • F.L.Tiplea: *Introduction to Cryptography* (in preparation) - chapters of the book will be available to students.

EVALUATION	conditions	
	criteria	
	evaluation methods	One project and a final exam.
	final result - formula	50% from the project and 50% from the final exam.

COURSE NAME	**MODELS OF SECURITY**	CODE: MSI1205

STUDY YEAR	I	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	2	-	-	56	184	8	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. CATALIN DIMA	LACL (Univ. Paris 12, France)

PREVIOUS COURSES REQUESTED	No prerequisite required.

OBJECTIVES	The course is an advanced introduction in the modeling,analysis and validation of security properties and systems. Students will achieve basic and intermediate-level knowledge of various security analysis techniques, rating from access control models to information flow models and models for security protocols ensuring confidentiality, authenticity, anonymity and/or game-like advresial situations like fair exchange and auctions.
GENERAL DESCRIPTION	The topics covered are: • A revision of classical access control models: discretionary/mandatory, role-based, type enforcement, etc.; specification of access control properties. • Models of information flow control: Bell-LaPadula, lattice models, models based on observability and interference, etc. • Specification of information flow properties in extensions of temporal logics. • Specifying information flow in programming languages, the Denning approach. • Models for security protocols: tree automata, multi-agent models, coalitions and game semantics, models for trust management. • Specification of various security properties (authenticity, anonymity, adversial and coalization situation) in extensions of temporal logics. • Computational models of security protocols.
DESCRIPTION OF SEMINARY / LABORATORY WORK	Seminars and laboratories are grouped around the chapter currently discussed in the course. They aim to illustrate the topics of the chapter mainly by practical applications.
TEACHING METHODS	On-line and blackboard presentation.

BIBLIOGRAPHY (SELECTION)	• Matt Bishop, *Computer Security, Art and Science*, Addison-Wesley, Pearson Education, 2002. • Christel Baier, Joost-Pieter Katoen, *Principles of Model Checking*, MIT Press, 2005. • Ronald Fagin, Joseph Halpern, Moshe Vardi, *Reasoning about Knowledge*, MIT Press, 2005. • Research papers on specific topics will be distributed during the lectures.

EVALUATION		
	conditions	
	criteria	
	evaluation methods	7 homeworks and a final exam.
	final result - formula	50% from the homeworks and 50% from the final exam.

COURSE NAME	**NETWORK SECURITY**	CODE: MSI1207

STUDY YEAR	I	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	184	8	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. FERUCIO LAURENȚIU ȚIPLEA	Computer Science

PREVIOUS COURSES REQUESTED	Familiarity with the basic cryptographic concepts is helpful, but can in principle be also acquired in parallel to attending the course.

OBJECTIVES	Provide students with an understanding of the themes and challenges of network security, the techniques for access control and intrusion detection, and the current state of the art. The students will have developed a critical approach of the analysis of network security, and will be able to bring this approach to bear on future decisions regarding network security. Practical skills will include the implementation of a security protocol.
GENERAL DESCRIPTION	The course covers four main topics: 6. Authentication applications 7. IP security 8. Electronic mail security 9. Web security
DESCRIPTION OF SEMINARY / LABORATORY WORK	Students will be required to complete a project in network security. These projects must be completed in small groups (no more than 3 students) and each group will be required to present their progress to the other students twice during the course.
TEACHING METHODS	On-line and blackboard presentation.

BIBLIOGRAPHY (SELECTION)	• William Stallings: *Cryptography and Network Security: Principles and Practice*, third ed., Prentice Hall, 2003. • Matt Bishop: *Computer Security: Art and Science*, Addison-Wesley Professional, 2002. • Matt Bishop: *Introduction to Computer Security*, Addison-Wesley, 2004. • Research articles and RFCs on IPsec, SSL-TLS, DNSsec, etc.

EVALUATION		
	conditions	
	criteria	
	evaluation methods	One project and a final exam.
	final result - formula	50% from the project and 50% from the final exam.

COURSE NAME	**SECURITY PROTOCOLS**	CODE: MSI1208

STUDY YEAR	I	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	184	8	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	DR. CONSTANTIN ENEA	Computer Science

PREVIOUS COURSES REQUESTED	Familiarity with basic cryptographic concepts as treated, for example, in the core course "Information Security", is helpful, but can in principle also be acquired in parallel to attending the course.

OBJECTIVES	The objectives of the course are to provide students with the practice of cryptographic protocols (efficient implementations, vulnerabilities, etc.) and to provide a background against which the student will be able to develop new protocols and applications.
GENERAL DESCRIPTION	Topics include: • basics on security (cryptographic) protocols • formalisms: MSR, strand spaces • undecidability of secrets • bounded protocols • taged protocols • recursive protocols • verification techniques: BAN,inductive method, strand spaces,based on I/O-automata
DESCRIPTION OF SEMINARY / LABORATORY WORK	All seminars will be oriented on the topic discussed during the courses. Students will be asked to prepare a research project.
TEACHING METHODS	On-line and blackboard presentation.

BIBLIOGRAPHY (SELECTION)	• F.L.Tiplea: Algebraic Foundations of Computer Science, Polirom, 2006. • F.L.Tiplea: *Introduction to Cryptography* (in preparation) - chapters of the book will be available to students. • Research articles.

EVALUATION	conditions	
	criteria	
	evaluation methods	One project and a final exam.
	final result - formula	50% from the project and 50% from the final exam.

COURSE NAME	**SOFTWARE SECURITY**	CODE: MSI2101

STUDY YEAR	II	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	184	8	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. GHEORGHE GRIGORAS	Computer Science

PREVIOUS COURSES REQUESTED	No prerequisite required.

OBJECTIVES	The course is an introduction in various programming-based methods for the development of security policies. Students will acquire experience with programming with various Java security-related packages and with access control modules in Linux.
GENERAL DESCRIPTION	The topics covered are: 10. Access control in Java 11. The Java Security and Cryptography extensions 12. SELinux, a Linux module for access control 13. Jif, a package for controlling information flow in Java programs 14. JAAS, the Java Authentication and Authorization System
DESCRIPTION OF SEMINARY / LABORATORY WORK	Seminars and laboratories are grouped around the chapter currently discussed in the course. They aim to illustrate the topics of the chapter mainly by practical applications.
TEACHING METHODS	On-line and blackboard presentation.

BIBLIOGRAPHY (SELECTION)	5. Scott Oaks, *Java Security*, O'Reilly, ISBN 978-0596001575 6. Frank Meyer, David Kaplan, Karl McMillan, *SELinux by Examples*, Prentice Hall PTR, ISBN 978-0131963696 7. Tutorial pages for JAAS at http://java.sun.com/javase/6/docs/technotes/guides/security/jaas/JAASRefGuide.html 8. Tutorial pages for Jif at http://www.cs.cornell.edu/jif/

EVALUATION		
	conditions	
	criteria	
	evaluation methods	7 homeworks and a final exam.
	final result - formula	50% from the homeworks and 50% from the final exam.

COURSE NAME	**WIRELESS AND MOBILE SECURITY**	CODE: MSI2102

STUDY YEAR	II	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	184	8	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	LECT. DR. SORIN IFTENE	Computer Science

PREVIOUS COURSES REQUESTED	Information Security

OBJECTIVES	The course will present the most important mechanisms dedicated to protect data integrity and confidentiality, access control, authentication, user privacy, quality and continuity of service,in wireless environments.
GENERAL DESCRIPTION	The topics of the course are: • Wireless technology overview • Risks and threats of wireless • Security under resource constraints (bandwidth, memory, computation, energy constraints) • Intrusion and anomaly detection in wireless environments • Key management in wireless environments • Privacy and anonymity in wireless environments • Public Key Infrastructure in wireless environments • Authentication, authorisation, and access controlin wireless environments • Standards in wireless security (Equivalent Privacy Standard (WEP), Extensible Authentication Protocol (EAP), Wi-Fi Protected Access (WPA, WPA2), IEEE 802.11i, Bluetooth 2.1) • Secure mobile commerce • Secure wireless multimedia broadcast
DESCRIPTION OF SEMINARY / LABORATORY WORK	Seminars intend to stimulate students in their own research in wireless security - improve their ability of extracting, presenting, and discussing results from themost relevant papers and try to extend/improve them.
TEACHING METHODS	On-line and blackboard presentation.

BIBLIOGRAPHY (SELECTION)	• E. Earle.Wireless Security Handbook, CRC Press, 2006. • N. Sklavos, X. Zhang. Wireless Security and Cryptography: Specifications and Implementations, CRC Press, 2007. • NIST Federal Information Processing Standards. • Conference and journal articles.

EVALUATION		
	conditions	presentation of a report on a selected topic (P), midterm exam (ME), final exam (FE)
	criteria	P, ME, FE ≥ 5
	evaluation methods	presentation of a report on a selected topic (P), midterm exam (ME), final exam (FE)
	final result - formula	0.4 P + 0.3 ME + 0.3 FE

COURSE NAME	**SECURITY OF OPERATING SYSTEMS**	CODE: MSI2104

STUDY YEAR	II	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
-	-	2	-	28	62	3	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	LECT. DR. CRISTIAN VIDRAŞCU	Computer Science

PREVIOUS COURSES REQUESTED	BSc in Computer Science

OBJECTIVES	This course offers an advanced introduction in the field of security of operating systems. The students who will attend this course will obtain knowledge about operating systems, regarding the techniques and mechanisms used for their security.
GENERAL DESCRIPTION	The course will treat the following subjects: • Basic notions. Security models. Identification and authentication. Authorization and accountability. • Access control in the system. Plocies used for access control: discretionary, mandatory, and role-based policies. Specification of access control policies. • The UNIX/Linux operating system. Security architecture. Classical access control mechanisms. Audit logs and intrusion detection methods. • The Windows operating system. Security architecture. Access control and audit mechanisms. Privilege elevation techniques. • Advanced access control techniques - SELinux framework, AppArmor, Flask security architecture. • Security evaluation standards: Orange Book,Common Criteria, etc. • Modern protection mechanisms: virtualization and sandboxing techniques, proof-carrying code.
DESCRIPTION OF SEMINARY / LABORATORY WORK	This is a research oriented course aiming to guide students in doing research in the field of security of operating systems.
TEACHING METHODS	Exposure using video-projetor, combined with explanations on blackboard and practical demos.

BIBLIOGRAPHY (SELECTION)	• Dieter Gollmann: *Computer Security*, John Wiley & Sons, 1999. • Matt Bishop, *Computer Security, Art and Science*, Addison-Wesley, Pearson Education, 2002. • William Stallings, Lawrie Brown: *Computer Security, Principles and Practice*, Prentice Hall, 2008. • Ross J. Anderson: *Security Engineering*, second edition, John Wiley & Sons, 2008. • Boris Loza: *Unix, Solaris and Linux: A Practical Security Cookbook*, Authorhouse Press, 2005. • Research papers on specific topics.

EVALUATION		
	conditions	
	criteria	
	evaluation methods	Practical labworks during the semester and final written test.
	final result - formula	Lab * 0.4 + WrittenThesis1 * 0.3 + WrittenThesis2 * 0.3

COURSE NAME	**MALICIOUS SOFTWARE**	CODE: MSI2104'

STUDY YEAR	II	SEMESTER	1	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
-	-	2	-	28	62	3	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	DR. CONSTANTIN ENEA	Computer Science

PREVIOUS COURSES REQUESTED	No requested prerequisite.

OBJECTIVES	This course aims to introduce students to the theory of malicious software (malware) such as computer viruses, worms, trojan horses, rootkits, spyware, dishonest addware, crimeware and other maliciousand unwanted software. Defense techniques, detection methods and cleaning technologies are alsodiscussed.
GENERAL DESCRIPTION	The course covers the following topics: 1. Introduction to malicious software (malware). Evolution of malware. 2. Computer viruses. Case study on (Brain, Adson, OneHalf, Bakaver, Evol). 3. Vulnerabilities. Classification of targeted systems. 4. Worms. Case studies on (Melissa, Sohanad, MyDoom, Slammer). 5. Trojan horses and their role in informatic attacks. 6. Bot-Nets. Fast-Flux methods. Malware as a a commercial purpose. Social engineering. 7. Detection methods. Anti-rootkit technologies. Defense techniques against malicious software. 8. System cleaning methods. Case study on (Vundo, Virut, NewDotNet). 9. Evaluating system safety.
DESCRIPTION OF SEMINARY / LABORATORY WORK	Research oriented on malicious software design.
TEACHING METHODS	On-line and blackboard presentation.

BIBLIOGRAPHY (SELECTION)	• Victor Oppelman, Oliver Friedrichs, Brett Watson: *Extreme Exploits. Advanced Defenses Against Hardcore Hacks*, McGraw-Hill Osborne Media, 2005. • James C. Foster, Vitaly Osipov, Nish Bhalla, Niels Heinen: *Buffer Overflow Attacks*, Syngress Media, 2005. • James C. Foster, Vincent Liu: *Writing Security Tools and Exploits*, Syngress Media, 2005. • James C. Foster, Mike Price: *Sockets, Shellcode, Porting & Coding*, Syngress Media, 2005. • Mark Burnett: *Hacking the Code*, Syngress Media, 2005. • Greg Hoglund, James Butler: *Subverting the Windows Kernel. Rootkits*, Addison-Wesley, 2006. • Research articles and software.

EVALUATION		
	conditions	
	criteria	
	evaluation methods	One project and final exam.
	final result - formula	50% from the project + 50% from the final exam.

COURSE NAME	**BELIEF LOGICS IN INFORMATION SECURITY**	CODE: MSI2205

STUDY YEAR	II	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	184	8	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. CRISTIAN-DUMITRU MASALAGIU	Computer Science

PREVIOUS COURSES REQUESTED	Familiarity with the basic cryptographic concepts is helpful, but can in principle be also acquired in parallel to attending the course.

OBJECTIVES	The objectives of the course are to provide students with basic analysis techniques for security protocols based on modal and beliefs logic, and to provide a background against which the student will be able to use software tools such as Isabelle and TLA+.
GENERAL DESCRIPTION	The topics of the course include: • logic: syntax, semantics, normal forms, decidability • logical theories, proof system, reasoning • modal logics • belief logics: BAN and GNY • applications and case studies in Isabelle and TLA+
DESCRIPTION OF SEMINARY / LABORATORY WORK	Seminars/laboratories will illustrate the concepts discussed during the course. Students will be also required to complete a project and to present their progress to the other students twice during the course.
TEACHING METHODS	On-line and blackboard presentation.

BIBLIOGRAPHY (SELECTION)	• C. Masalagiu: Fundamentele logice ale informaticii, Ed. Univ. "Al.I.Cuza", Iaşi, 2004. • M. Huth, M. Ryan: Login in Computer Science, Cambridge Univ. Press, 2006. • E.M. Clarke, O. Grumberg, D.A. Peled: Model Checking, MIT Press, 1999. • L. Lamport: Specifying Systems: The TLA+ Language and Tools for Hardware and Software Engineers, Person Edication, 2002. • W.S. Cooper: The Evolution of Reason. Logic as a Branch of Biology, Cambridge Univ. Press, 2001. • Research papers on BAN and GNY logics (to be explicitely mentioned during the course)

EVALUATION		
	conditions	Any student has toprove that he/she actually assisted to the labs.
	criteria	Any mentioned activity has to be quoted.
	evaluation methods	One project, activity during laboratories, and a final exam.
	final result - formula	50% from the project and activity during laboratories and 50% from the final exam.

COURSE NAME	**SECURITY OF ELECTRONIC COMMERCE**	CODE: MSI2206

STUDY YEAR	II	SEMESTER	2	COURSE STATUS (**C**-compulsory/**OP**-optional/**F**-facultative)	C

HOURS PER WEEK				TOTAL HOURS PER SEMESTER	TOTAL HOURS INDIVIDUAL ACTIVITY	CREDITS	EVALUATION (**P**-during the semester, **C**-oral examination, **E**-written examination, **M**-mixed)	TEACHING LANGUAGE
C	S	L	Pr.					
2	-	2	-	56	184	8	M	Romanian

COURSE TEACHER	TEACHING AND SCIENTIFIC DEGREE, FIRST NAME, LAST NAME	DEPARTMENT
	PROF. DR. VICTOR PATRICIU	Technical Military Academy, Bucharest

PREVIOUS COURSES REQUESTED	Facultative: Applied Cryptography

OBJECTIVES	This course introduces students to the challenge of electronic commerce and Business on Internet, vulnerabilities and defenses.
GENERAL DESCRIPTION	The course covers the following topics: 1. digital signatures and certificates and PKI 2. smart-cards, biometrics and payment systems 3. e-commerce security 4. reglementations in e-commerce
DESCRIPTION OF SEMINARY / LABORATORY WORK	PGP, cryptographic libraries (BSAFE, Open SSL, MS CAPI/CAPICOM, Cryptolib, Java Cryptography), SET (Secure Electronic Transaction), iKP (Internet Keyed Payments), eCash, NetCash. Students will also be involved in writing a Java Card application for a dual digital signature under SET.
TEACHING METHODS	On-line and blackboard presentation.

BIBLIOGRAPHY (SELECTION)	1. V. Patriciu, I. Bica, M. Pietrosanu, I. Priescu, Semnaturi electronice si securitate informatica, Ed. All, 2005. 2. V. Patriciu, I. Bica, M. Pietrosanu, C. Vaduva. N. Voicu, Securitatea comertului electronic, Ed. All, 2001. 3. V. Patriciu, S. Patriciu, I. Vasiu, Internetul si dreptul, Ed. All, 1999. 4. Mostafa Hasem Sherif, Protocole for Secure Electronic Commerce, CRC Press, 2004. 5. C. Radu, Implementing Electronic Card Payment Systems, Artech House Computer Security Series, 2003. 6. W. Stalling, Cryptography & Network Security, Prentice Hall, 2001. 7. D. O'Mahony, Electronic Payment Systems for E-Commerce, Artech House, 2001. 8. R. Housley, Planning for PKI, John Wiley, 2000. 9. W. Ford, Secure Electronic Commerce, Prentice Hall, 2001.

EVALUATION		
	conditions	
	criteria	
	evaluation methods	7 small projects and a final exam.
	final result - formula	50% from the projects + 50% from the final exam.

III. OTHER INFORMATION ABOUT THE FACULTY

1. Rooms

- 5 lecture rooms
- 4 seminary rooms
- 8 laboratories (20-25 computers/room)

2. Student associations

The Association of Computer Science Students in Iaşi (ASII)
- E-mail: `asii@infoiasi.ro`
- Web: `http://students.infoiasi.ro/~asii`

3. Industry partnerships

The Faculty of Computer Science has currently developped partnerships with important IT firms, such as:

- Siemens Romania, who offers every years a certain amount of student scholarships for carrying out the practice stage within the firm
- Microsoft Romania
- Continental VDO (former Siemens VDO) Romania
- DiTech Romania
- Embarcadero Romania
- Code40
- Synygy

4. Student facilities

The library

The students of the Faculty of Computer Science have access to both the Central University Library "Mihai Eminescu" and its faculty branch (i.e., the Library of the Faculty of Computer Science).

The collection of the Library of the Faculty of Computer Science comprises more than 2000 volumes (books and journals) in the fields of computer science and mathematics. Besides, the library grants access, on an online reservation basis, to the collection of around 100 installation and documentation CDs provided by Microsoft Romania.

Internet access

The students of the Faculty of Computer Science have free Internet access from the faculty's laboratories. These are open on weekdays (Monday to Friday), between 8-20, and are meant both for seminary/laboratory classes and for students' individual training.

Internet access is also available from the campuses of the "Alexandru Ioan Cuza" University.

Accomodation

For students who do not live in Iaşi, the "Alexandru Ioan Cuza" University provides accomodation in its campuses: Titu Maiorescu, Codrescu, Târguşor Copou.

Scholarships

The scholarship system applied by the Faculty of Computer Science complies with the specific regulations of the "Alexandru Ioan Cuza" University. The main categories are study and performance scholarships (granted to students with excellent learning results) and social support scholarships (granted to students with lower income). There are also a series of scholarship categories that encourage and reward various kinds of performance.

CONTENTS