University of Split

Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture

PROPOSAL FOR THE GRADUATE STUDY PROGRAMME

Computer Science

Split, March 15, 2005

STUDY PROGRAMME

Graduate Study: Computer Science

Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture
Address: Ruđera Boškovića b.b., HR-21000 Split
Phone: + 385 21 305777
Fax: + 385 21 463877
dekanat@fesb.hr
http: //www.fesb.hr

1. Introduction

1.1. General information on the programme

Computer science, as both scientific and engineering discipline, deals with the research as well as the use of information, focusing on the design and changing of structures that communicate, store and process information. Computer science is related nowadays with a wide variety of fields of human activities. The basic concepts show a high level of similarity whether it deals with hardware and software systems or natural or social systems. There is a growing need for specialists of such a profile and encompasses the need for expert use of ready-made solutions through the design, application and use of very complex systems up to the original research work in the field of computer science and interdisciplinary areas overlapping with it.

Present situation and a projection of the development point out a great need for specialists in computer science. Only sufficient number of well-educated professionals in this academic discipline is the basic prerequisite for the implementation of aims defined in the "Strategy of the Development of the Republic of Croatia in the 21st century".

All areas of the human activity have been strongly influenced by general computerization and a rapid development of new technologies. Everybody needs a computer for various purposes and many will wish to study at least some forms of computer science. Computer engineering will remain to be present in enhancing careers of many professionals. Those who choose to pursue the path of computer engineering will have a significant role in the future forming of society. For further development of modern society, it is very important that the study of Computer Science attracts students with different interests and prepares them to be capable and responsible experts.

The proposed Computer Science study programme aims at educating and training professionals in the field of computer science and upon completion of studies graduates will be qualified for work in industry, higher education and other public and state institutions.

Estimation of the study objectives in relation to the labour market

Split is an industrial and university centre important for the whole region of Dalmatia and neighbouring surroundings. FESB is the only faculty in this region, which offers the university study programme of Computer Science. In a short-term period, the labour market in Croatia will lack several thousand experts from the field of Computer sciences whereas the European Union will need a hundred times that number. The frequent contacts between enterprises in the Split region and neighbouring surroundings confirm these estimations. As a result, this profile of experts seems to be very promising which is approved by the growing number of students enrolling every year in this study programme. Successful completion of the study programme will enable the graduates in Computer Sciences to take on jobs requiring skills in the industry, computer and communication companies, higher education,

services, etc. It is to point out that there is no place, where a graduate in Computer Sciences would not be successful in his work. Consequently, the need for these experts in today's labour market is enormous. It is of particular importance in these days when social and economic changes demand the development of new, small or medium, technologically advanced companies that will be a new support to the economical development.

Connection of the study curriculum with modern trends in computer science profession

The study programme has been designed to provide the computer sciences student with basic theoretical knowledge and practical skills qualifying them for work in the industry as well as acquiring new knowledge and new technologies. Teaching methods and techniques to be implemented within the study programme encourage critical thinking and creativity, which are essential to students' professional development. Students will be encouraged to be independent in decision-making and will be particularly trained for individual and team-work. The study programme, based on modern concepts of syllabus design in Computer Sciences, follows up world and in particular European trends in higher education as well as economical needs. In accordance to that, appropriate study programmes are being designed. In all strategically important documents concerning Croatia and EU computer-information technologies are listed as one of priorities, which opens up a path for taking part in home, European and world projects. The development of computer technologies is based, along with the progress in the field of computer science itself (e.g. cryptography, grid and embedded systems), on the latest insights into natural sciences (e.g. quantum information sciences and nanotechnology), clearly emphasizing interdisciplinary character of the study and correlation to modern scientific understanding. Scientific cooperation with renowned foreign research institutions is a basic orientation of FESB: Scientists working at FESB actively contribute to the development of computer science and computer technology.

Comparability with study programmes of higher education institutions abroad

The study programme has been designed following the model of other respected high education institutions in Europe. The best practices of American universities have been adopted and resumed in the document "Computing Curricula 2004" of the leading professional associations if the field of computer sciences (The Association for Computing – ACM, The Association for Information Systems – AIS, The Computer Society – IEEE-CS). There are no two countries in the world that share the same education system. This refers particularly to type and organization of study, duration of study, study title, professional or academic title or degree on completion of the study. In general, mathematics and basic natural sciences are given priority, and are followed by basic engineering and information sciences courses as well as specialist courses related to specific branches of computer sciences. General courses have also been implemented. The proposed study programme has been adjusted in accordance with recommendations of ERASMUS and THEIERE projects ('Towards the Harmonisation of Electrical and Information Engineering Education in Europe', http:/www.eaeeie.org/theiere/). The programme proposal has been designed in conformity with recommendations of SEFI (European Society for Engineering Education) and CESAER (Conference of European Schools for Advanced Engineering Education and Research). Electric Engineering Studies are comparable to other study programmes in several renowned European universities as:

- Techniche Univerzität Wien, Austria
- Eidgenössische Technische Hochschule (ETH) Zürich, Switzerland

1.2. Previous experience in the field

FESB has long tradition in the organisation of lectures with similar programs. The Faculty of Electrical Engineering was established in the town of Split in 1960. The Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture (FESB) was founded in 1971 and in 1974 the Faculty became a part of the University of Split. Vocational studies were established in 1979 and have been continuing to date with intermission of three years (1998-2001). Within the graduate study programme in Electrical Engineering computer engineering was introduced in 1985. More than 200 students have graduated at that course so far.

A full study of Computer Science was introduced in 2001 due to the need for a larger number of professionals in this specialization area. More than 700 students have been enrolled in the Computer Science study so far; first students are expected to graduate in the first semester of the academic year 2005.

The FESB reputation has been contributed by numerous presentations of research results obtained by professors and young researches at home and abroad. We also point out the contribution of graduate engineers, former students of this Faculty, who have become renown professionals working in various countries all over the world.

1.3. Student mobility scheme

The study programme in Computer Science enables vertical and horizontal mobility of students. As to vertical mobility the graduate study programme in Computer Science is mainly open to related postgraduate study programmes. As to horizontal mobility, it is open to the mobility of students between related studies of Croatian universities. Students will be allowed to finish a part of their study programmes at similar institutions at home or abroad.

Students educated at the study of Computer Science at FESB will gain sufficient knowledge to take up jobs concerning design, implementation and use of computer systems in a broader technical field, as well as in other fields requiring more complex computer systems. Therefore, mobility is encouraged, leaving students the choice to select courses offered at other faculties at home or abroad.

On the other hand, the need for specialists in computer engineering is in increase in all disciplines, leaving therefore the study open to all students of other study programmes who can gain further education through elective courses.

1.4. Other elements

FESB provides all conditions for the realisation of the study of Computer Science. The advantage of the programme is an optimal utilization of teaching stuff, facilities and equipment. The Faculty employs sufficient number of lecturers and other staff having suitable scientific and professional qualification, fully in conformity with the standards prescribed for the teaching work-load by the law and collective agreement for the higher education. The suitable premises and equipment in conformity with the needs of high quality level of studying are also provided. Aditionnaly, the program is supported not only by entrepreneurs and public sector of the Split Dalmatian County, but also by the wider area comprising Dalmatian region and state authorities. FESB has signed the agreements on Collaboration in promotion of scientific and educational activities and they realise joint projects with the whole range of

organizations belonging to the economic and public sector: Split-Dalmatian County, City of Split, Ministry of Defence, Split Shipyard, Croatian Electrical Company, Croatian Telecom, Croatian Academic and Research Network – CARNet, Ericsson Nikola Tesla, Technological Center Split, Brodosplit, Siemens, VIPnet, Microsoft Croatia, Airport Split, TLM Šibenik.

2. General description

Type of programme	Graduate	
Programme title	Computer Science	
Institution	**Proposed by**	Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture
	Participating institutions	Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture
Duration	2 years	
ECTS	120	
Admission requirements	Completed undergraduate studies (BS) in Computer Science or some other related studies and appropriate enrolment procedure. Minimum ECTS credits:180	
Learning outcomes and competences	After successful completion of graduate studies, the Master in Electrical Engineering will be qualified in the design, implementation and maintenance of complex computer systems with emphasis on the integration of software and hardware solutions. Successful completion of graduate studies will enable him keep pace with modern scientific achievements and technological changes.	
Access to further studies	After successful completion of graduate studies the Master of Science will be able to continue doctoral studies within the research field of Computer Studies or some other related doctoral studies.	
Qualification awarded	Academic Degree: Master's Degree in Computer Science Academic Title: Master of Science (MS) in Computer Science	

3. Study programme

3.1. Programme structure with credits

1st Semester			
Course code	**Course title**	**Course structure *L+E+P**	**ECTS**
	Numerical analysis	30+30+0	5
	User interfaces	30+0+30	5
	Computing models	30+0+30	5
	Introduction to artificial intelligence	30+0+30	5
	Computer graphics	30+0+30	5
	Elective	30+0+30	5
Total:		**180+30+150= 360**	**30**
*L = Lecture, E = Exercise, P = Practice (Laboratory)			

2nd Semester			
Course code	**Course title**	**Course structure *L+E+P**	**ECTS**
	Languages and compilers	30+0+30	5
	Optimisation methods	30+0+30	5
	Advanced computer architecture	30+0+30	5
	Multimedia systems	30+0+30	5
	Distributed databases	30+0+30	5
	Elective	30+0+30	5
Total:		**180+0+180= 360**	**30**
*L = Lecture, E = Exercise, P = Practice (Laboratory)			

3rd Semester			
Course code	**Course title**	**Course structure** ***L+E+P**	**ECTS**
	Cryptography and network security	30+0+30	5
	Grid computer systems	30+30+0	5
	Business information systems	30+0+30	5
	Embedded systems	30+0+30	5
	Elective 1 **	30+0+30	5
	Elective 2 **	30+0+30	5
Total:		**180+30+150= 360**	**30**
*L = Lecture, E = Exercise, P = Practice (Laboratory)			

Elective courses			
Course code	**Course title**	**Course structure** ***L+E+P**	**ECTS**
	Data compression	30+0+30	5
	Advanced algorithms	30+30+0	5
	Quantum informatics	30+0+30	5
	Cryptography	30+0+30	5
	Digital comunnications	30+0+30	5
	Datawarehouse	30+0+30	5
	Business intelligence	30+0+30	5
	Project management	30+0+30	5
	Software agents	30+0+30	5
*L = Lecture, E = Exercise, P = Practice (Laboratory)			

4th Semester			
Course code	**Course title**	**Course structure** ***L+S+E+P**	**ECTS**
	Master thesis	0+15+0+0	30
Total:		**0+15+0+0=15**	**30**

3.2. Course information

Course title	**Numerical Analysis**		
Course code			
Type of course	Applied mathematics course		
Level of course	Advanced		
Year of study		**Semester/trimestar**	
ECTS (Number of credits allocated)	5 ECTS (Lectures and exercises 30+30 hours – 1.5 ECTS. Study, exams and homeworks - 3.5 ECTS.)		
Name of lecturer	Prof. dr. sc. Ivan Slapnicar		
Learning outcomes and competences	Students shall acquire knowledge from numerical analysis: error analysis, numerical integration, numerical solutions of systems of nonlinear equations and numerical solutions of ordinary differential equations. The students will be competent in numerically solving practical problems arising in natural sciences and engineering.		
Prerequisites	Mathematical analysis, basic programming course.		
Course contents	Error analysis: floating point arithmetics, backward error, forward error. Numerical integration: Peano's Kernel Theorem, Gaussian integration, Romberg algorithm, Euler-Maclaurin summation formula, adaptive integration. Systems of nonlinear equations: bisection, Newton method, Fixed-point theorem. Numerical solutions of ordinary differential equations: single-step and multi-step methods, Runge-Kutta methods, solutions of boundary value problems.		
Recommended reading	D. Kincaid, W. Cheney, Numerical Analysis-Mathematics of Scientific Computing, Brooks/Cole Publishing Company, 2002. V. Hari at all, Numerical Analysis, PMF-MO, Zagreb, 2003, (in Croatian). D. N. Arnold, A Concise Introduction to Numerical Analysis, University of Minnesota, Minneapolis, 2001.		
Supplementary reading	J. Stoer, R. Bulirsch, Introduction to Numerical Analysis, Springer, New York, 1993.		
Teaching methods	Classsroom teaching, computer lab exercises, assignements for individual and group work.		
Assessment methods	Homeworks, seminar works and final exam.		
Language of instruction	Croatian, English is possible on demand.		
Quality assurance methods	Student feedback via questionnaries and surveys. Consultations with the head of the study program. Evaluation of teaching by the agency for quality promotion.		

Course title	**User Interface Design**		
Course code			
Type of course	Theoretical and practical Elective		
Level of course	Basic level course		
Year of study		Semester	
ECTS (Number of credits allocated)			
Name of lecturer	Assist. Prof. Dr. Andrina Granić		
Learning outcomes and competences	Human computer interaction stresses the importance of good user interfaces and the relationship of interface design to effective human interaction with computers. This course offers: - theoretical knowledge and practical experiences in the fundamental aspects of designing, implementing and evaluating user interfaces - understanding what is meant by 'good design', as well as by designing usable interactive computer systems - knowledge of a variety of simple methods for an interface quality evaluation		
Prerequisites	No formal prerequisites are required, although is assumed that students have basic computer knowledge.		
Course contents	Fundamental theoretical knowledge and practical experiences in designing, implementing and evaluating user interfaces of interactive computer systems, here including: definition of domain and basic concepts, understanding users and their tasks, design principles and guidelines, user-centered interface development, usability engineering, usability evaluation methods (with or without users), techniques for prototyping, as well as for graphical user interface implementation.		
Recommended reading	- J. Preece, Y. Rogers, H. Sharp, D. Benyon, S. Holland and T. Carey: *Human-Computer Interaction*, Addison-Wesley, Harlow, England, 1994. - J. Nielsen: *Usability Engineering*, AP Professional, Boston, 1993. - D. Norman: *The Psychology of Everyday Things*, Basic Books, 1988 - A. Granić: *Osnove i principi interakcije čovjeka i računala*, Fakultet prirodoslovno-matematičkih znanosti i odgojnih područja, Sveučilište u Splitu, http://www.pmfst.hr/~granic/		
Supplementary reading	- J. Preece, Y. Rogers and H. Sharp: *Interaction Design: Beyond Human-Computer Interaction*, John Wiley & Sons, 2002. - R. M. Baecker, J. Grudin, W. Buxton and S. Greenberg: *Readings in Human-Computer Interaction: Toward the Year 2000*, 2nd Ed., Morgan Kaufmann Publishers, San Francisco, CA, 1995.		
Teaching methods	Lectures, practical session, research project.		
Assessment methods	Written/practical exam and oral exam, seminar paper presentation.		
Language of instruction	Croatian / English		

Quality assurance methods	Students' assesment, tachers and field experts evaluation.

Course title	**Computing models**		
Course code			
Type of course	Obligatory		
Level of course	Advanced level course		
Year of study	third	**Semester/trimestar**	summer
ECTS (Number of credits allocated)	6 ECTS (lectures 30 hours - 1 ECTS / exercises 15 hour - 0,5 ECTS / laboratory 15 hours - 0,5 ECTS / tests, consultations, laboratory practice, seminar and learning 120 hours - 4 ECTS)		
Name of lecturer	dr.sc. Julije Ožegović, associate.prof		
Learning outcomes and competencies	Course provides fundamental theoretical knowledge of automata, grammars and languages as computer science core basis.		
Prerequisites	None		
Course contents	Programmable machines. Program description of algorithms. Language processors. Lexical, syntax and semantics analysis. Intermediate code generation. Formal definition of alphabet, string and language. Operations between languages. Regular languages: Finite automata. Regular expressions. Forma grammars. Regular grammars. Context free languages: Context free grammar. Push down automata. Recursively countable languages: Turing machine. Grammar of unbounded productions. Context dependent languages: Context dependent grammar. Linear bounded automata. Language, automata and grammar taxonomy: Structural complexity of languages. Complexity of language recognition.		
Recommended reading	1. Srbljić, Siniša.: Jezuični procesori 1, Element, Zagreb, 2002.		
Supplementary reading			
Teaching methods	Lectures / Exercises course / Laboratory practice / Seminar / Tutorials		
Assessment methods	Continuous assessment: laboratory tests, practical tests, knowledge tests, seminar presentation. Exam: written and oral as unity.		
Language of instruction	Croatian		
Quality assurance methods	Student feedback via questionnaires and surveys Occassional class observations by Head of Department		

Course title	**Introduction to artificial intelligence**		
Course code			
Type of course	**Lecture / Seminar** / Exercise Course **Obligatory** / Elective / Optional		
Level of course	**Basic level course** Intermediate level course Advanced level course Specialised level course		
Year of study	4	**Semester/trimestar**	7
ECTS (Number of credits allocated)	5 ETCS 30 hours lecture, 30 hours laboratory work, 30 hours student project, 60 hours learning for test		
Name of lecturer	Prof.dr.sc.Darko Stipaničev – lecture Ljiljana Bodrožić, dipl.ing. - seminar		
Learning outcomes and competences	Essential knowledge of artificial intelligence, theory, programming languages and applications.		
Prerequisites	Main knowledge of computers and programming.		
Course contents	Intelligence and artificial intelligence. Problem solving – methods and principles. Knowledge elicitation and storage (mathematical logic, semantic networks., frames, stereotypes, scenarios, production rules) Search (blind search, directed search, breth first search, deep first search, best first search, hill climbing, A* algorithm). Artificial intelligence languages (Prolog & Lisp). Artifical intelligence applications.		
Recommended reading	- A.Cawsey, The Essence of Artificial Intelligence, Prentice Hall, 1998. - S.Russel, P.Norvig, Artificial Intelligence: A Modern Approach, Prentice Hall, 2nd Ed. 2002.		
Supplementary reading	- AI on the Web (http://http.cs.berkeley.edu/%7Erussell/ai.html) - American Association for Artificial Intelligence (www.aaai.org)		
Teaching methods	**Lectures** / Seminars (discussion, seminar papers, reports) / Exercise course / **Tutorials / Practical session** / Distance learning / Advisory hours / Project work/ Research project		
Assessment methods	Continuous assessment (diagnostic tests, independent homework, project tasks, achievement tests, seminar papers ...) Exam: **written** / oral / seminar paper presentation		
Language of instruction	Croatian		
Quality assurance methods	Student feedback via questionnaires and surveys Lectures responsible for the same subject area collaborate closely and monitor each other's work. Occasional class observations and appraisal by Head of Department		

Course title	**Computer graphics**		
Course code	XXX???		
Type of course	Lecture / Seminar / Exercises on PC Obligatory		
Level of course	Advanced level course		
Year of study	4	**Semester/trimestar**	7
ECTS (Number of credits allocated)	5 (30 hours lectures and 30 hours exercises ~ 2 ECTS, seminar work ~ 30 hours - 1 ECTS, individual learning ~ 60 hours - 2 ECTS)		
Name of lecturer	Dr.sc.Vladan Papić, assistant professor		
Learning outcomes and competences	Understanding basic aspects of computer graphics. Student is capable to develop and apply computer graphics algorithms in C, and also familiar with the use of graphical libraries in programming.		
Prerequisites	Basic programming skills		
Course contents	Introduction. Basic algorithms of raster graphic. Grafical hardware. Geometrical transformations. Objects in 3D. Curves and surfaces. Rendering. OpenGL. Animation.		
Recommended reading	1. V.Papić, Računalna grafika, internal textbook. 2. Foley, Computer Graphics: Principles and Practice (second edition in C), Addison-Wesley Publishing Company, 1996.		
Supplementary reading	3. Rogers, Procedural Elements of Computer Graphics, McGraw-Hill Science/Engineering/Math; 2nd edition, 1997.		
Teaching methods	Lectures and PC exercises (30+30). Lectures with the use of audio-visual equipment and PC. PC exercises are using following applications: Visual C++, SGRP, OpenGL		
Assessment methods	Continuous assesment. Exam prerequisites: finished and accepted seminar, exercises. Exam: oral.		
Language of instruction	Lectures in Croatian. Literature is available in English.		
Quality assurance methods	Student feedback via questionnaries and surveys Lectures responsible for the same subject area collaborate closely and monitor each other's work. Occassional class observations and appraisal by Head of Department		

Course title	**Compilers and Formal Languages**		
Course code			
Type of course	Lecture / Execise Course Obligatory		
Level of course	Advanced level course		
Year of study	4	**Semester/trimestar**	2
ECTS (Number of credits allocated)	5		
Name of lecturer	Ivo Mateljan		
Learning outcomes and competences	In this course you will learn about modern compiler techniques. This course has two main goals. One is to introduce you to formal languages, automata theory, and software design. The other is to clarify the fundamental concepts behind language design and implementation.		
Prerequisites			
Course contents	Automata Theory: regular expressions, finite automata, context free grammars, PDAs. Deterministic parsing: LR parsers and the flex/bison tools. Semantic analysis: abstract syntax trees, symbol tables, type checking, resource allocation. Virtual machines and run-time environments: stacks, heaps, objects. Code generation: resources, templates, optimizations.		
Recommended reading	Ivo Mateljan: Programski prevodioci, on-line skripta, FESB, 1988. Aho, Sethi, Ullman: Compilers - Principles, Techiques and Tools, Adison Wesley, 1986.		
Supplementary reading	Allen I. Holub: Compiler design in C, Prentice Hall, 1990.		
Teaching methods	Lectures / Seminars (discussion, seminar papers, reports) / Exercise course / Practical session / Advisory hours		
Assessment methods	Continuous assesment (diagnostic tests, independent homework, project tasks, achievement tests, seminar papers ...) Exam: oral / seminar paper presentation		
Language of instruction	Croatian		
Quality assurance methods	Student feedback via questionnaries and surveys Lectures responsible for the same subject area collaborate closely and monitor each other's work. Occassional class observations and appraisal by Head of Department		

Course title	**Optimization methods**		
Course code			
Type of course	Elective		
Level of course	Advanced level course		
Year of study	5	**Semester/trimestar**	2
ECTS (Number of credits allocated)	5 ECTS (lectures 30 hours - 1 ECTS / exercises 30 hours - 1 ECTS / tests and exam 5 hours, consultations and learning 85 hours – 3 ECTS)		
Name of lecturer	dr.sc. Jadranka Marasović, asist. prof.		
Learning outcomes and competencies	Obtained basic theoretical knowledge about different methods used for fastest and organized way of finding optimal solutions. Obtained practical knowledge with the aim of solving simple problems independently and digital computer client oriented use.		
Prerequisites	None		
Course contents	Introduction: quantitative and qualitative models. Graph theory. Function criteria. Constraints. Linear programming. Dual solutions and sensitivity analysis. Integer programming. Nonlinear programming: 1D and nD methods with and without constraints, gradient method. Heuristic programming – methods and use. Transport problem. Shortest path problem. Location and distribution problems. Dynamic programming. Inventory problems. Decision theory and game theory approach. Principles of operations research in technical, biological, economic, social systems. Operations research and digital computer client oriented use.		
Recommended reading	1. Belegundu, Chandrupatla: Optimization Concepts and Applicatins in Engineering, Prentice Hall, 1999. 2. T.B. Boffey: Graph Theory in Operations Research, McMillan Press, Hong Kong, 1982. 3. R. Bronson, G. Naadimuthu: Operations Research, Schaum's Outline of Operations Research, McGraw Hill, 1998. 4. H.A. Taha: Operations Research: An Introduction, Prentice Hall, 1997.		
Supplementary reading	1. J. Marasović: Uvod u operacijska istraživanja, Interna skripta, FESB, Split, 2000.		
Teaching methods	Lectures / Seminars/Exercises course / Tutorials		
Assessment methods	Continuous assessment: practical tests, knowledge tests. Exam: oral.		
Language of instruction	Croatian		
Quality assurance methods	Student feedback via questionnaires and surveys Lectures responsible for the same subject area collaborate closely and monitor each other's work. Occasional class observations by Head of Department		

Course title	**Advanced computer architecture**		
Course code			
Type of course	Lecture / Seminar / Execise Course Obligatory / Elective		
Level of course	Specialised level course		
Year of study	4	**Semester/trimestar**	7
ECTS (Number of credits allocated)	5		
Name of lecturer	Pro. dr. sc. Sven Gotovac		
Learning outcomes and competences	Understand the advanced computer architecture.		
Prerequisites	Computer architecture		
Course contents	Parallel computing principles. MIMD. SIMD. Vector architecture. Superscalar computers. Tidily coupled systems. Hardly coupled systems. Scalability. Memory consistency. Connection networks. Benchmarking. Latency. Bandwidth. Improving memory performance. NUMA. COMA. Node intercommunication. Synchronization between processors. Scheduling. Load balancing. Tools for parallel system software development. Parallel system operational systems and compilers. PVM. MPI. POSIX Threads. Thread Marks.		
Recommended reading	Ribarić, S.: Naprednije arhitekture mikroprocesora, Tehnička knjiga, Zagreb		
Supplementary reading	Hennesy & Patterson, "Computer Architecture: A Quantitative Approach", 3rd edition, Morgan Kaufmann		
Teaching methods	Lectures / Seminars (discussion, seminar papers, reports) / Exercise course / Project work		
Assessment methods	Continuous assessment: diagnostic tests, independent homework, project tasks, seminar papers. Exam: oral / seminar paper presentation		
Language of instruction	Croatian		
Quality assurance methods	Student feedback via questionnaries and surveys Lectures responsible for the same subject area collaborate closely and monitor each other's work. Occassional class observations and appraisal by Head of Department		

Course title	**Multimedia systems**		
Course code			
Type of course	Lecture / Exercise Course Obligatory / Elective		
Level of course	Specialised level course		
Year of study	721/4, 750/4	**Semester/trimestar**	7
ECTS (Number of credits allocated)	5 (30 hours lecture + 30 hours exercise course + 60 hours studying)		
Name of lecturer	dr.sc. Hrvoje Dujmić, doc.		
Learning outcomes and competences	Basic theoretic knowledge in multimedia systems and virtual reality systems which includes characteristics of speech, audio, image and video signals (including 3D images and video). Theoretical foundation of efficient compression of above mentioned signals. Practical knowledge which includes algorithms for compression of the above mentioned signals.		
Prerequisites	First year undergraduate.		
Course contents	History of multimedia systems; Media (text, images/graphics, audio/speech, animation/video, interaction); Multimedia authoring tools; Multimedia application design; Multimedia applications; Computer/Human Multimedia communications; Analog/Digital conversion; Sampling frequency; Audio and speech signal characteristics; Speech and audio formats (wav, mod, mp3, aif, ...); Audio and speech compression; Speech specific algorithms (LPC, CELP, coding in mobile telephony); Audio specific algorithms (mp3); Basic color theory; Color models; Image signal characteristics; Image formats (gif, tiff, jpg, bmp, ...); Image compression (JPEG, Wavelets); TV and video signals characteristics; Video signal formats (mpg, avi, asf, ...); Video compression (H.261, H.263, H.26L, MPEG 1, MPEG 2, MPEG 4): Basic of virtual reality systems (3D images, 3D video, 3D audio), Multimedia network communications		
Recommended reading	1. Hrvoje Dujmić: "Multimedia systems and Virtual reality systems", FESB Split, interna skripta Rao, Bojkovic, Milovanovic: "Multimedia Communication Systems: Techniques, Standards and Networks", Prentice Hall, 2002		
Supplementary reading	1. Steinmetz, Nahrstedt: "Multimedia Fundamentals: Media Coding and Content Processing", Prentice Hall, 2002 2. IEEE Transaction on Multimedia IEEE Multimedia Magazine		
Teaching methods	Lectures / Exercise course Advisory hours		
Assessment methods	Continuous assessment (diagnostic tests) Alternatively: written exam		
Language of instruction	Croatian		
Quality assurance methods	Student feedback via questionnaires and surveys Lectures responsible for the same subject area collaborate closely and monitor each other's work. Occasional class observations and appraisal by Head of Department.		

Course title	**Distributed Databases**		
Course code			
Type of course	Lecture / Seminar / Execise Course Obligatory		
Level of course	Advanced level course		
Year of study	5	**Semester/trimestar**	7
ECTS (Number of credits allocated)	5		
Name of lecturer	Hrvoje Dujmić, Marko Šušnjar		
Learning outcomes and competences	Principles and practice of implementing distributed databases, including design, query processing and transaction processing.		
Prerequisites			
Course contents	The principles and system organization of distributed databases. Data fragmentation and distribution, distributed database design, query processing and optimization, distributed concurrency control, reliability and commit protocols, and replicated data management. Distributed algorithms for data management: clocks, deadlock detection, and mutual exclusion. Heterogeneous and federated distributed database systems. Overview of commercial systems and research prototypes.		
Recommended reading	Principles of Distributed Database Systems, M. Tamer Ozsu, Patrick Valduriez Database Design and Development, Paulraj Ponniah		
Supplementary reading			
Teaching methods	Lectures / Seminars (discussion, seminar papers, reports) / Exercise course / Practical session / Advisory hours / Research project		
Assessment methods	Continuous assesment (diagnostic tests, independent homework, project tasks, achievement tests, seminar papers ...) Exam: oral / seminar paper presentation		
Language of instruction	Croatian		
Quality assurance methods	Student feedback via questionnaries and surveys Lectures responsible for the same subject area collaborate closely and monitor each other's work. Occassional class observations and appraisal by Head of Department		

Course title	**Cryptography and Network Security**		
Course code			
Type of course	Lecture / Exercise Course Obligatory / Elective		
Level of course	Specialised level course		
Year of study		**Semester/trimestar**	
ECTS (Number of credits allocated)	5 (30 hours lecture + 30 hours exercise course + 60 hours studying)		
Name of lecturer	dr.sc. Hrvoje Dujmić, doc.		
Learning outcomes and competences	Basic theoretic knowledge about mathematics foundation of cryptography . Practical knowledge about the most important cryptography methods including applications in the network security.		
Prerequisites	Undergraduate study.		
Course contents	history of cryptography, classical encryption techniques, Cezar, Vigenère, Playfair, Hill; modern block ciphers, Data Encryption Standard (DES), Advanced Encryption Standard (AES), Public-key cryptography, RSA algorithms, other public-key cryptosystems, message authentication and hash function, digital signature, authentication application (Kerberos, X.509), electronic mail security, IP security, web security, standards and standards-setting organizations		
Recommended reading	W. Stallings: Cryptography and Network Security. Principles and Practice, Prentice Hall, 2003. D. R. Stinson: Cryptography. Theory and Practice, CRC Press, 2002		
Supplementary reading	M. Welschenbach: Cryptography in C and C++, Apress, Berkeley, 2001 B. Schneier: Applied Cryptography, John Wiley, 1995 A. J. Menezes, P. C. Oorschot, S. A. Vanstone: Handbook of Applied Cryptography, CRC Press, 1996.		
Teaching methods	Lectures / Exercise course Advisory hours		
Assessment methods	Continuous assessment (diagnostic tests) Alternatively: written exam		
Language of instruction	Croatian		
Quality assurance methods	Student feedback via questionnaires and surveys Lectures responsible for the same subject area collaborate closely and monitor each other's work. Occasional class observations and appraisal by Head of Department.		

Course title	**Grid computing systems**		
Course code			
Type of course	Lecture / Seminar / Execise Course Obligatory		
Level of course	Advanced level course		
Year of study	5	**Semester/trimestar**	9
ECTS (Number of credits allocated)	5		
Name of lecturer	Dr. sc. Sven Gotovac, redoviti profesor, mr. sc. Eugen Mudnić		
Learning outcomes and competences	Understanding grid system. How to develop grid application.		
Prerequisites			
Course contents	Grid systems evolution, grid applications, grid infrastructure, grid computing organizations, grid standards , grid computing anatomy, grid computing models, grid data storage models, autonomic computing, service oriented grid arhitecture, web-service arhitecture, OGSA (open grid service arhitecture), OGSA platform components, OGSI (open grid services infrastructure), grid middleware, grid computational economy		
Recommended reading	Grid Computing, Joshy Joseph, Craig Fellenstein , Prentice Hall, 2003		
Supplementary reading	Grid Computing, F. Berman, G. Fox, T. Hey, Wiley, 2003		
Teaching methods	Lectures / Seminars (discussion, seminar papers, reports) Tutorials / Practical session / Advisory hours / Research project (za diplomski)		
Assessment methods	Continuous assesment (diagnostic tests, independent homework, project tasks, achievement tests, seminar papers ...) Exam: seminar paper presentation		
Language of instruction	Croatian		
Quality assurance methods	Student feedback via questionnaries and surveys Lectures responsible for the same subject area collaborate closely and monitor each other's work. Occassional class observations and appraisal by Head of Department		

Course title	**Business Informatin Systems (BIS)**		
Course code			
Type of course	Lecture / Seminar / Execise Course Obligatory		
Level of course	Intermediate level course		
Year of study	5	**Semester/trimestar**	3
ECTS (Number of credits allocated)			
Name of lecturer	dr.sc. Stipe Celar		
Learning outcomes and competences	The main goals are to provide graduates with the relevant skills to take a responsibility for a success of using IT-systems in a whole business systems.		
Prerequisites	Finished 4rd year		
Course contents	The most important focus is on: • BIS Lifecycle • subsystems and arts of BIS and • business organization's maturity with a brief hitory of BIS (from business applications to EAS – *Enterprise Application Suite*). Some *dilemmas* and actual questions will be discussed: • centralised *vs*. distributed information system (IS) • build *vs*. buy • "right" tehnology – "right" operating systems – "right" database management system. Very important phases of BIS-lifecycle are: • the acquisition phase – it must define a gap between business processes and a chosen solution • BIS implementation – many projects fails in this phase. Overview of the market segments and the top EAS solutions will be given. Some solutions will be exercised in seminars. Advanced BIS (KM – Knowledge Management, BI –Business Intelligence, CI – Competitive Intelligence) will be introduced.		
Recommended reading	www.gartner.com; www.metagroup.com; www.sei.cmu.edu; www.idc.com		
Supplementary reading	www.microsoft.com; www.sap.com www.oracle.com		
Teaching methods	Lectures / Seminars (discussion, seminar papers, reports) / Exercise course / Tutorials / Practical session / Distance learning / Advisory hours / Research project		
Assessment methods	Continuous assesment (diagnostic tests, independent homework, project tasks, achievement tests, seminar papers ...) Exam: writen / oral / seminar paper presentation		
Language of instruction	Croatian		
Quality assurance methods	Student feedback via questionnaries and surveys Lectures responsible for the same subject area collaborate closely and monitor each other's work. Occassional class observations and appraisal by Head of Department		

Course title	**Embedded systems**		
Course code			
Type of course	Lecture / Seminar / Execise Course Obligatory		
Level of course	Advanced level course		
Year of study	5	**Semester/trimestar**	3
ECTS (Number of credits allocated)	5		
Name of lecturer	Prof. dr. sc. Sven Gotovac		
Learning outcomes and competences	Principles of operation and design of embedded systems. Embedded software an operating systems. Real time OS. Hardware-software codesign.		
Prerequisites	Computer architecture Operating systems		
Course contents	Interfacing of microcomputers to peripherals or other computers for purposes of data acquisition, device monitoring and control, and other communications. The interfacing problem is considered at all levels including computer architecture, logic, timing, loading, protocols, and software laboratory for building and simulating designs. Design of embedded digital systems; microcontrollers, embedded programs, real-time operating systems, design methodologies, hardware-software codesign, hardware modeling and computer-aided design.		
Recommended reading	**Wayne Wolf, Computers as Components Principles of Embedded Computing Systems Design**, Morgan Kaufmann 2001., 1-55860-541-X Frank Vahid, Tony D. Givargis, **Embedded System design: A Unified Hardware/Software Introduction,** John Wiley 2001, ISBN 0-471-38678-2		
Supplementary reading			
Teaching methods	Lectures / Seminars (discussion, seminar papers, reports) / Exercise course / Tutorials / Practical session / Distance learning / Advisory hours / Research project (za diplomski)		
Assessment methods	Continuous assesment (diagnostic tests, independent homework, project tasks, achievement tests, seminar papers ...) Exam: writen / oral / seminar paper presentation		
Language of instruction			
Quality assurance methods	Student feedback via questionnaries and surveys Lectures responsible for the same subject area collaborate closely and monitor each other's work. Occassional class observations and appraisal by Head of Department		

Course title	**Data compression**		
Course code			
Type of course	Lecture / Exercise Course Elective		
Level of course	Specialised level course		
Year of study	5	**Semester/trimestar**	
ECTS (Number of credits allocated)	5 (30 hours lecture + 30 hours exercise course + 60 hours studying)		
Name of lecturer	dr.sc. Hrvoje Dujmić, doc.		
Learning outcomes and competences	Theoretic knowledge about data compression. Practical knowledge about the most important data compression methods.		
Prerequisites	Undergraduate study.		
Course contents	Mathematical preliminaries for data compression (source model, source coding, Shannon lower bound, rate-distortion theoy), losless compression, lossy compression, vector and scalar quantization, (optimality, LBG algorithm), transform and predictive coding, probability, probability coding, Huffman method, Shannon-Fano method, arithmetic codig, dictionary coding, adaptiv coding, run length coding, Ziv i Lempel (LZ77, LZ78), LZW, LZSS, LZH algorithms, data compression (fax, zip, text, images, JBIG, JPEG-LS), data compression standards		
Recommended reading	Khalid Sayood: "Introduction to Data compression", Morgan Kaufmann Publishers, 2000		
Supplementary reading			
Teaching methods	Lectures / Exercise course		
Assessment methods	Continuous assessment (diagnostic tests) Alternatively: written exam		
Language of instruction	Croatian		
Quality assurance methods	Student feedback via questionnaires and surveys Lectures responsible for the same subject area collaborate closely and monitor each other's work. Occasional class observations and appraisal by Head of Department		

Course title	**Advanced algorithms**		
Course code			
Type of course	Lecture / Exercise Course Elective		
Level of course	Specialised level course		
Year of study		**Semester/trimestar**	
ECTS (Number of credits allocated)	5 (30 hours lecture + 30 hours exercise course + 60 hours studying)		
Name of lecturer	dr.sc. Hrvoje Dujmić, doc.		
Learning outcomes and competences	Develop ability to design efficient algorithms. Develop ability to prove correctness and evaluate efficiency of algorithms. Study important algorithm areas and techniques not normally covered in the course Algorithms.		
Prerequisites	Algorithms. Data structure.		
Course contents	Linear Programming, Network Flows, Approximation Algorithms, Planarity Testing of Graphs, Number-Theoretic, Randomized techniques, Routing techniques in networks, Algorithms for the construction of effective compilers, String handling for DNA-analysis, Sorting while minimizing time and space, Amortized analysis, String matching algorithms, Algorithms for NP-hard and NP-complete problems, cryptographic algorithms, parallel algorithms		
Recommended reading	Hrvoje Dujmić: „Napredni algoritmi“, interna skripta T.Cormen, C.Leiserson, R.Rivest, C.Stein: „Introduction to Algorithms“, second edition, third printing, McGraw-Hill, 2002		
Supplementary reading	Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin. Network Flows: Theory, Algorithms, and Applications. Prentice Hall, 1993. Vijay Vazirani. Approximation Algorithms. Springer-Verlag, 2001		
Teaching methods	Lectures / Exercise course		
Assessment methods	Continuous assessment (diagnostic tests) Alternatively: written exam		
Language of instruction	Croatian		
Quality assurance methods	Student feedback via questionnaires and surveys Lectures responsible for the same subject area collaborate closely and monitor each other's work. Occasional class observations and appraisal by Head of Department		

Course title	**Quantum Informatics**		
Course code			
Type of course	Lecture / Exercise Course Elective		
Level of course	Intermediate level course		
Year of study	1	**Semester/trimestar**	1
ECTS (Number of credits allocated)	5 (30 hours lectures – 2 ECTS, 30 hours excercises – 2 ECTS, 30 hours homework 1 ECTS)		
Name of lecturer	Ivica Puljak, PhD		
Learning outcomes and competences	Understanding concepts of modern physics and modern technology and its application to real life problems.		
Prerequisites	Physics I and Physics II or similar courses		
Course contents	*Introduction to quantum mechanics*: De Broglie hypotesis, Heisenberg uncertainty relations, Schrödinger wave equation, Potential barrier, Potential well, Tunnel effect, Postulates of Quantum Mechanics. *Quantum computing*: Quantum devices, Quantum algorithms. *Physical realisation of quantum computers*: Optical quantum computers, Ion traps, Nuclear magnetic resonance. Quantum information: entanglement, quantum cryptography, superdense coding, quantum teleportation, quantum noise and errors, quantum errors correction.		
Recommended reading	1. Nielsen, M. A.; Chuang, I. L.; *Quantum Computation and Quantum Information*, Cambridge University Press, 2000. 2. Bouwmeester, D. ; Ekert, A.; Zeilinger, A.; *The Physics of Quantum Information*, Springer 2004. 3. Gershenfeld, N.: *The Physics of Information Technology*, Cambridge University Press, 2000		
Supplementary reading	1. Ho-Kim, Q.; Kumar, N.; Lam, C. S.: *Invitation to Contemporary Physics*, Worlds Scientific, 2004.		
Teaching methods	Lectures / Practical session / Advisory hours		
Assessment methods	Continuous assesment (achievement tests) Exam: oral		
Language of instruction	Croatian and english if needed		
Quality assurance methods	Student feedback via questionnaries and surveys. Lectures responsible for the same subject area collaborate closely and monitor each other's work.		

Course title	**Digtital Communications**		
Course code			
Type of course	Lecture /Execise Course Obligatory /		
Level of course	Advanced level course		
Year of study	5	**Semester/trimestar**	
ECTS (Number of credits allocated)	6		
Name of lecturer	Nikola Rožić		
Learning outcomes and competences	▪ Understanding of real channel influences on digital signal transmission, adaptive signal processing for minimum distorsion, ▪ Syncronization aspects and methods in digital communications, ▪ optimal detection methods, error detection and correction methods, ▪ hard nad soft decoding, iterative (turbo) decoding and MAP decisions, ▪ understanding and knowledge of performance of systems with automatic repeat request (ARQ) mechanisms, ▪ understanding of secure coding (cryptography) and conditional access methods, ▪ understanding and knowledge of different network topologies and network architectures for continous and burst like traffic, ▪ model of communication switching center and signalling protocols, ▪ knowledge of switching systems technologies, time (T) i time-space-space (TST) switching, switching methods for packet networks, ▪ traffic theory and information model of circuit-switched and packet-switched networks.		
Prerequisites			
Course contents	Real channels, intersymbol interference, Nyquist filters, correlative filters, linear and nonlinear equalization, optimal transmitting and receiving filters, echo cancelling, adaptive filtering and equalization, pseudorandom coding (scrambling), parallel and serial transmission, synchronous and asynchronous communications, simplex, half duplex and duplex communications, synchronization (bit, frame and carrier), channel coding, block and convolution coding, BCH, RS, turbo codes, trellis codes, ARQ systems and FEC systems, ciphering and security, network topologies, switching center model, circuit switching and packet switching, multistage switching systems, TST and STS structures, X.25 networks, flow control, ISDN networks and protocols, signalling No.7, mobile wireless networks, ATM protocols, LAN protocols, X.400 and TCP/IP protocols, heterogeneous networks, IP telephony, traffic engineering and traffic flow modelling.		
Recommended reading	1. Rožić, N.: Digital Communications (interna skripta) 2. J.R.Barry, E.A.Lee, D.G.Messerschmitt: Digital Communication, III. Ed.		
Supplementary reading	3. A.Bažant i drugi: Osnovne arhitekture mreža, Zagreb, 2003. 4. I.A.Clover, P.M.Grant: Digital Communications, Prentice Hall 1998.		
Teaching methods	Lectures / Exercise course / Practical session / Advisory hours / Project work (za preddiplomski)/ Research project (za diplomski)		
Assessment methods	Continuous assesment (diagnostic tests, independent homework, project tasks, achievement tests, seminar papers ...) Exam: writen / oral / seminar paper presentation		
Language of	Croatian/English		

instruction	
Quality assurance methods	Student feedback via questionnaries and surveys Lectures responsible for the same subject area collaborate closely and monitor each other's work. Occassional class observations and appraisal by Head of Department

Course title	**Data Warehouse**		
Course code			
Type of course	Lecture / Seminar / Execise Course Elective		
Level of course	Advanced level course		
Year of study	5	**Semester/trimestar**	8
ECTS (Number of credits allocated)	5		
Name of lecturer	Marko Šušnjar		
Learning outcomes and competences	This course provides the knowledge and skills to create a Data Warehouse model from an OLTP (On-Line Transaction Processing) system and then design the dimensions and facts for meeting the requirements of the OLAP (On-Line Analytical Processing) system or Data Mart.		
Prerequisites			
Course contents	Introduction and review of data modeling. Understand dimensional modeling. Understand dimension outriggers, degenerate dimensions and other variations in the data warehouse model. OLTP vs. OLAP. Star Schemas and snowflake designs. Dimensions and Facts Creating Fact Tables Outrigger Dimensions		
Recommended reading	Building the Data Warehouse, W. H. Inmon, W. H. Inmon Data Warehouse Design Solutions, Christopher Adamson, Michael Venerable		
Supplementary reading			
Teaching methods	Lectures / Seminars (discussion, seminar papers, reports) / Exercise course / Practical session / Advisory hours / Research project (za diplomski)		
Assessment methods	Continuous assesment (diagnostic tests, independent homework, project tasks, achievement tests, seminar papers ...) Exam: oral / seminar paper presentation		
Language of instruction	Croatian		
Quality assurance methods	Student feedback via questionnaries and surveys Lectures responsible for the same subject area collaborate closely and monitor each other's work. Occassional class observations and appraisal by Head of Department		

Course title	**Buisness Intelligence**		
Course code			
Type of course	Lecture / Seminar / Execise Course Elective		
Level of course	Specialised level course		
Year of study		**Semester/trimestar**	
ECTS (Number of credits allocated)			
Name of lecturer	Marko Šušnjar		
Learning outcomes and competences	By the end of this course students will have a firm grasp of the concepts behind knowledge management technologies. Students will be able to evaluate an organizations use of KM technologies and be able to tell where and when different KM technologies are appropriate in an organization.		
Prerequisites			
Course contents	This course provides an overview of information technologies and trends that support knowledge management. The basic systems for storage, access, and retrieval of information, and specific systems that organize information, including knowledge bases, webs, nets, groupware, decision support systems, and collaboration environments, will be explored and discussed, as will the process of knowledge management in enterprise-wide and globally networked environments.		
Recommended reading	Poslovna inteligencija, Željko Panian, Goran Klepac Business Intelligence Roadmap: The Complete Project Lifecycle for Decision-Support Applications, Larissa T. Moss, Shaku Atre Business Intelligence for the Enterprise, Mike Biere		
Supplementary reading			
Teaching methods	Lectures / Seminars (discussion, seminar papers, reports) / Exercise course / Practical session / Advisory hours / Research project (za diplomski)		
Assessment methods	Continuous assesment (diagnostic tests, independent homework, project tasks, achievement tests, seminar papers ...) Exam: oral / seminar paper presentation		
Language of instruction	Croatian		
Quality assurance methods	Student feedback via questionnaries and surveys Lectures responsible for the same subject area collaborate closely and monitor each other's work. Occassional class observations and appraisal by Head of Department		

Course title	**Project management**		
Course code			
Type of course	Lecture / Exercise Course / Seminar / Obligatory / Elective / ???		
Level of course	Specialised level course		
Year of study	?	**Semester/trimester**	?
ECTS (Number of credits allocated)	**4** ECTS (2+0+1) Lectures: 30 hours: **1** ECTS Exercise: 15 hours: **0,5** ECTS Project: 30 hours **1** ECTS Preparation for exam: 45 hours: **1,5** ECTS		
Name of lecturer	Ph.D. Boženko Bilić, associated professor		
Learning outcomes and competences	Competences for projects planning and managing.		
Prerequisites	-		
Course contents	Introduction. Project management basics. Projects – vision, strategy, goals (examples: automobile industry and shipbuilding industry). Basics of organization. A project organization. Simultaneous Engineering. Project planning methods. Quality management (quality planning, improvement and control) – QFD (Quality Function Deployment), FMEA (Failure Mode and Effect Analysis). Examples. Time and resource planning: Gantt charts, network-based techniques – PERT (Project Evaluation and Review Technique, CPM (Critical Path Method), PRECEDENCE. Examples. Target Costing. Continuous improvement – Kaizen. Psycho-social component of project management. Creativity stimulation methods.		
Recommended reading	1. NN: A Guide to the Project management Body of Knowledge (PMBOK) --- 2000 Edition, Project Management Institute, 2001. 2. Veža, I.: Upravljanje projektima (interna skripta), Fakultet elektrotehnike, strojarstva i brodogradnje, Split, 2002. 3. Kerzner, H.: Project Management: A Systems Approach to Planning, Scheduling, and Controlling, John Wiley & Sons, New York, ISBN: 0471225770, 2003 4. Wysocki, R. K.; McGary, R.: Effective Project Management: Traditional, Adaptive, Extreme, John Wiley & Sons, New York, ISBN: 0471432210, 2003		
Supplementary reading			
Teaching methods	Lectures / Seminars (discussion, seminar papers, reports) / Exercise course / Tutorials / Advisory hours		
Assessment methods	Exam: / oral and seminar paper presentation		
Language of instruction	Croatian.		
Quality assurance methods	Student feedback via questionnaires and surveys. Occasional class observations and appraisal by Head of Department.		

Course title	**Software agents**		
Course code			
Type of course	Lecture Elective		
Level of course	Advanced level course		
Year of study		**Semester/trimester**	
ECTS (Number of credits allocated)	5		
Name of lecturer	Prof. Dr. sc. Darko Stipaničev, Mr. sc. Maja Štula		
Learning outcomes and competences	Agent technology, agents, multiagent systems, agent oriented software design basic knowledge.		
Prerequisites	Skills obtained after completion of second year study.		
Course contents	Agents. Agents example, intelligent agents, agents and objects. Mathematical formalization of agents. Types and agent architectures. Multiagent systems. Areas of multiagent systems deployment, organization and interoperation definition. Speech act theory. Agent communication language,. Communication definition and models. Interaction protocols. Knowledge representation, ontology, content languages. Agent management architecture, agent platforms.		
Recommended reading	Ferber J., *Multi-agent Systems, An Introduction to Distributed Artificial Intelligence*, *Addison-Wesley*, England, 1999. Wooldridge M., Jennings N., *Intelligent Agents: Theory and Practice*, Knowledge Engineering Review, Vol. 10, No. 2, Cambridge University Press, 1995		
Supplementary reading	Wais G. (Editor): *Multiagent Systems*, *MIT Press*, Cambridge, Massachusetts, 1999. Jennings N.R., Wooldridge M.J., *Applications of Intelligent Agents*, Agent Technology: Foundations, Applications and Markets, pp. 3-28, 1998.		
Teaching methods	Lectures / Seminars (discussion, seminar papers, reports) / Exercise course / Tutorials / Practical session / Advisory hours / Project		
Assessment methods	Exam: written and seminar paper presentation		
Language of instruction	Croatian		
Quality assurance methods	Student feedback via questionnaires and surveys Lectures responsible for the same subject area collaborate closely and monitor each other's work. Occasional class observations and appraisal by Head of Department		

Course title	**Master thesis**		
Course code			
Type of course	Guided personal study.		
Level of course	Advanced level course		
Year of study	II	**Semester**	IV
ECTS (Number of credits allocated)	30 Number of allocated credits is based on estimation showing that the student needs 850 hours for the thesis research and preparation and 50 hours for the preparation of the oral presentation.		
Name of lecturer	Lecturer from the selected subject.		
Learning outcomes and competences	After the Master thesis is completed, the learner is expected to acquire knowledge he/she evaluated in collaboration with the mentor within the selected subject.		
Prerequisites	Completed all courses of 3rd semester of graduate study.		
Course contents	The student selects the subject of the Master thesis according to the previously defined subjects determined by the Faculty Council for each academic year. The Student performs individual and independent research in the subject selected in collaboration with the lecturer/mentor. The Student accomplishes her/his Master thesis in written form.		
Recommended reading	According to the subject lecturer recommendation.		
Supplementary reading	According to the subject lecturer recommendation.		
Teaching methods	Consultations with selected subject lecturer and individual research work, as well as accomplishment of the Master thesis in a defined form.		
Assessment methods	Oral presentation of the Master thesis.		
Language of instruction	Croatian.		
Quality assurance methods	Quality assurance will be performed at three levels: (1) University level; (2) Faculty level by Quality Control Committee; (3) Lecturer's level.		