QUALIFICATION CHARACTERIZATION

OF MAJOR FIELD OF STUDY "INFORMATICS"

FOR "BACHELOR OF SCIENCE" DEGREE

WITH PROFESSIONAL QUALIFICATION "**INFORMATICIAN**"
4 YEARS (8 SEMESTERS)

I. Requirements to professional qualities and competences of students, completed this major field of study

Neofit Rilski South-Western University prepares qualified experts in Informatics that can apply their knowledge and skills in the area of science, culture, education and economics in Bulgaria and abroad.

After completion of Bachelor of Science (BSc) degree in Informatics, they can successfully realize themselves as: programmers, system and network administrators and designers, graphic designers, scientists, experts in hardware and software technologies.

At completion of Bachelor of Science degree in Informatics, students obtain:

- ✓ profound knowledge in the area of Informatics;
- ✓ good preparation in the area of Informatics and Mathematics as well as solid practical skills conforming to modern European standards and requirements;
- ✓ good opportunities for realizing as experts in Bulgaria or abroad;
- ✓ thinking style and affinity to the quickly changing requirements of the information society;
- ✓ opportunity for successful continuation of education in higher degrees (Master of Science and PhD) in Bulgaria and abroad.

II. Requirements to preparation of students completing this major field of study

Students completed BSc degree in Informatics have to possess following knowledge, skills and competences:

- ✓ to adapt and introduce program products and systems;
- ✓ to take part in development of program products and packages;
- ✓ to use mathematical models and software packages for solving real economic, engineering and management problems in continuous and discrete macrosystems;
- ✓ to solve various optimization problems;
- ✓ to use computer systems for automating the production process and management.

Qualification characterization of Major field of study "Informatics" for BSc degree is a basic document that determines rules for developing the curriculum. This qualification characterization is conformed with legislation in the area of higher education in Republic of Bulgaria.

CURRICULUM
Field of Study: Computer Science

First Year			
First Semester	ECTS credits	Second Semester	ECTS credits
Compulsory Courses		***Compulsory Courses***	
Mathematical Analysis 1	8.5	Mathematical Analysis 2	8.5
Computer mathematics 1	9	Computer mathematics 2	6
Introduction to Computer Programming	10	Mathematical Logic	5
Foreign Language 1	2.5	Object-Oriented Programming	8
Sport	0	Foreign Language 2	2.5
		Sport	0
	Total 30		Total 30
Second Year			
First Semester	ECTS credits	Second Semester	ECTS credits
Compulsory Courses		***Compulsory Courses***	
Discrete Mathematics	9	Mathematical Optimization	8
Computer Programming and Data Structures	5.5	Operating Systems	7
Differential Equations and Applications	5.5	Optional 2	4
Computer Architectures	5	Optional 3 (and 4)	11
Optional 1	5	Sport	0
Sport	0	***Optional Courses*** (1 course)	
		Data analysis with MS Excel and VBA	4
Optional Courses (1 course)		Programming with .NET Framework	4
Number Theory		Graph Theory	4
Mathematical Analysis 3		Introduction in LATEX 2ε	4
Discrete Functions		Graphic design of printed and promotional materials	4
Special Matrices		Programming with Ruby	4
Introduction in information systems and technologies		***Optional Courses*** (1 course with 11 credits or 2 courses with 5.5 credits)	
Development of object-oriented applications with design patterns		Combinatorics, Coding Theory and Cryptography	11
Software Quality Assurance		Mathematical Models of Economics	5.5
Introduction to XML		Matroid Theory	5.5
		Separable Sets of Variables	5.5
	Total 30		Total 30

Third Year			
First Semester	ECTS credits	Second Semester	ECTS credits
Compulsory Courses		***Compulsory Courses***	
Functional Programming	4	Logical Programming	5
Numerical Analysis	8	Probability and Statistics	8
Specialized statistical software	5.5	Databases	8
Theoretical Fundamentals of Computer Science	5.5	Algorithms in Graphs and Networks	7
Crises and Disasters	2	Optional 6	2
Optional 5	5	***Optional Courses*** (1 course)	
		Practical Course in Databases	
Optional Courses (1 course)		Practical Course in Logical Programming	
Operations Research		Workshop on asynchronous and parallel programming with the .NET Framework	
Mathematical Fundamentals of Computer Graphics		Practical Course in Web Design	
Object Pascal and Delphi Programming		Bitwise operations, graphs and combinatorial applications	
Mathematical Theory of Database		Management and Finance of Educational and Scientific Programs	
C++ Builder Programming		Design and development of Human Computer Interactions	
JavaScript programming		Norms and Standards of Information Security	
Domain Specific Languages		Web Content Management	
		Language culture	
	Total 30		Total 30

Fourth Year			
First Semester	ECTS credits	Second Semester	ECTS credits
Compulsory Courses		***Compulsory Courses***	
Computer Networks and Communications	6.5	Software Engineering	7
Artificial Intelligence	6	Optional 9	6.5
Internet Programming	6.5	Optional 10	6.5
Optional 7	5.5	Preparation of Undergraduate Thesis or	10
Optional 8	5.5	Preparation for State Exam	
Optional Courses (2 courses)		***Optional Courses*** (2 courses)	
Applied Statistics		Computer Models in Natural Sciences	
Expert Systems		Pattern Recognition	
Numerical Optimization		Internet Technologies	
Object-Oriented and Distributed Databases		Computer Security	
Multimedia Databases		Computer Design	
Algorithms for Decision Making in Management and Economics		Development of Database Applications	
		E-trading and corporate information systems	
Application for mobile devices		Information retrieval and web search	
Interactive Multimedia Technologies		Methods and tools for computer systems integration	
NoSQL Databases			
Metadata			
XML Standards for File Formats of MS Office			
	Total 30		Total 30

TOTAL FOR 4 ACADEMIC YEARS: 240 CREDITS

MATHEMATICAL ANALYSIS 1

Semester: 1

Course Type: lectures and tutorials

Hours per Week/FS/SS: 2 lecture hours and 2 tutorial hours per week/FS

ECTS Credits: 8.5 credits

Course Status: Compulsory Course in the Computer Science B.S. Curriculum

Course Description: The main topics to be considered:

- Numerical sequences
- Numerical Series
- Limit, continuity and differentiability of functions
- Integrals of functions of real variables
- Applications of the integral calculation

Course Aims: This course develops in details the problems of numerical sequences, numerical series, differential and integral calculation of functions of one real variable.

Teaching Methods: lectures, tutorials, consultations, homework, problem-solving, tests. During the lectures, students are acquainted with basic theoretical material – definitions, theorems, applications, with the methods of theorem proofs. During seminars, students solve practical problems. The knowledge obtained within the theoretical practice is used and it is also used in the process of problem solving.

Requirements/Prerequisites: Basic knowledge of courses in Elementary Mathematics, Linear Algebra, Analytical Geometry is necessary.

Assessment: Written exam on seminars and discussion on the theoretical material from the lectures.

Registration for the Course: not necessary

Registration for the Exam: Students and the lecturer agree on the convenient dates within the announced calendar schedule of examination session.

References:

Basic Titles:

1. V.A. Ilin, V.A. Sadovnichi, B.H. Sendov, "Mathematical Analysis", vol. 1 and 2, Science and Art, Sofia, 1989. (in Bulgarian).

2. Ya. Tagamlitzky, "Differential Calculus", Science and Art, Sofia, 1971 (in Bulgarian).

3. Ya. Tagamlitzky, "Integral Calculus", Science and Art, Sofia, 1971 (in Bulgarian).

4. I. Prodanov, N. Hadjiivanov, I. Chobanov, "Collection of Problems of Differential and Integral Calculus", Science and Art, Sofia, 1976 (in Bulgarian).

Additional Titles:

1. S.M. Nikolskii, "Course of Mathematical Analysis", vol. 1 and 2, Science, Moscow, 1973 (in Russian).

2. L.D. Kudrjavtcev, "Mathematical Analysis", vol. 1 and 2, Science, Moscow, 1976 (in Russian).

Abbreviation: FS: Fall Semester SS: Spring Semester

COMPUTER MATHEMATICS 1

Semester: 1

Type of Course: lectures and tutorials in computer lab

Hours per week - 2 hours lectures and 2 hours tutorials in computer lab/winter semester

Credits Numbers: 6 credits

Course Status: Compulsory course.

Course description:

It includes basic concepts, principles and methods in the field of calculus, linear algebra and analytical geometry, and their practical applications, also possibility to apply new IT in this area.

Objectives:

- The students should obtain knowledge of:
- To apply the methods of calculus, linear algebra and analytical geometry in practice
- To realize concrete applications with tools of IT.

Methods of teaching: seminars, tutorials, discussions, project based method.

Pre- requirements: Mathematics and IT

Assessment and Evaluation

Projects- 75%

Final Test- 25%

The course is successful completed with at least 50% of all scores.

Registration for the Course: required

Registration for the Exam: coordinated with the lecturer and the Student Service Office

INTRODUCTION IN PROGRAMMING

Semester: 1

Type of Course: **lectures, seminars and tutorials in computer lab**

Hours per week - **4 hours lecture, 1 seminar, and 2 hours tutorials in computer lab/**

Credits Numbers: **10 credits**

Course Status: Core course in curriculum of major Informatics, Bachelor degree.

Course description: Introduction to programming is the first course in scope of programming for the students of major "Informatics". The course includes topics of syntax and semantics of programming languages construct and statements. The course is based on the C++.

Objectives: The main goal of the course is the students to master principles of programming and algorithms.

Methods of teaching: lectures, tutorials, discussions, problem passed method, Project based method.

Pre- requirements: No need.

Assessment and Evaluation

Practical work and test- 50%

Final Exam 50%

The course is successful completed with at least 65% of all scores.

Registration for the Course: No

Registration for the Exam: coordinated with the lecturer and the Student Service Office

References

1. Магдалина Тодорова, **Програмиране на** C++, (първа и втора част), Сиела, София, 2002

2. Herbert Schildt, C++: **The Complete Reference,** Third Edition, McGraw Hill, 1998

3. Presentations at http://www.e-learning.swu.bg

ENGLISH LANGUAGE 1

Semester: 1
Course type: Seminars
Hours (weekly): 2 hours per week/FS

Number of ECTS credits: 2,5

Type of the course in the curriculum:

Compulsory course from the curriculum of the "Informatics" Bachelor's degree programme
Course description:

Introducing students to the basic components of English phonology, morphology and syntax. It helps students learn and practice communicating in everyday situations including asking and answering questions, using the telephone, taking messages, initiating conversations, asking for directions, making invitations and closing conversations. Class activities include role-playing, small-group activities and short presentations. It also develops skills in reading speed and comprehension. Students are introduced to reading strategies such as skimming, scanning, guessing meaning from context, previewing, predicting, making inferences and

giving opinions. Reading materials include short stories, news articles, computer passages and a simplified novel.

Goal:
The goals of the course is to enable students to speak and write effectively and confidently in their professional and personal lives. Students become acquainted with the basic terminology in the specific field.

Teaching methods: Seminars
Prerequisites: The knowledge acquired at high school is useful.
Examination and assessment procedures: The estimation of the acquired knowledge is based on a written exam

Course enrolment: students should submit an application at the academic affairs department at the end of the current semester
Registration for examination: coordinated with the lecturer and the academic affairs department
References:

1. English for Mathematicians and Computer Scientists, Sofia 2001
2. Raymond Murphy, English Grammar in Use I, Prosveta, Sofia, 1994
3. Raymond Murphy, English Grammar in Use II, Prosveta, Sofia, 1994
4. Soars, John & Liz, New Headway Elementary, Oxford University Press, 2000
5. Soars, John & Liz, New Headway Pre-Intermediate, Oxford University Press, 2007
6. Ранкова, М., Иванова, Ц., Английска граматика, Наука и изкуство, София, 1998

Notes: FS: Fall semester SS: Spring semester

MATHEMATICAL ANALYSIS 2

Semester: 2

Course Type: lectures and tutorials

Hours per Week/FS/SS: 2 lecture hours and 2 tutorial hours per week/SS

ECTS Credits: 8.5 credits

Course Status: Compulsory Course in the Computer Science B.S. Curriculum

Course Description: The course in Mathematical Analysis 2 includes basic concepts of mathematical analysis: improper integrals, functions of two and more variables, continuity of functions of several variables, partial derivatives, local and constrained extremum, implicit functions, double and triple Riemann integrals and their applications for finding arias and volumes, line integrals of first and second type, surface integrals of first and second type, basic formulas for integrals of Mathematical Physics.

Course Aims: Students should obtain knowledge for Mathematical Analysis 2, which is a basic mathematical course. This knowledge is necessary for studying Mathematical Analysis-3, Ordinary Differential Equations, Numerical Methods, Optimization.

Teaching Methods: lectures and tutorials.

Requirements/Prerequisites: Mathematical Analysis 1.

Assessment: Written final exam, two problem solving tests per semester.

Registration for the Course: not necessary

Registration for the Exam: Coordinated with the lecturer and Student Service Department.

References:

Basic Titles:

1. V.A. Ilin, V.A. Sadovnichi, B.H. Sendov, "Mathematical Analysis", vol. 1 and 2, Science and Art, Sofia, 1989. (in Bulgarian).

2. Ia. Tagamlitzky, "Differential Calculus", Science and Art, Sofia, 1971 (in Bulgarian).

3. Ia. Tagamlitzky, "Integral Calculus", Science and Art, Sofia, 1971 (in Bulgarian).

Additional Titles:

4. I. Prodanov, N. Hadjiivanov, I. Chobanov, "Collection of Problems of Differential and Integral Calculus", Science and Art, Sofia, 1976 (in Bulgarian).

Abbreviation:

FS: Fall Semester

SS: Spring Semester

COMPUTER MATHEMATICS 2

Semester: 2 semester

Course Type: lectures and lab exercises

Hours per Week/FS/SS: 2 lecture hours and 2 lab hours per week/FS

ECTS Credits: 6 credits

Course Status: Compulsory Course

Course Description: The course in Computer mathematics 2 includes basic mathematical methods of Analysis and Algebra, Relations and Functions and their properties, Elements of General Algebra, Combinatorics, Number Theory, Graph Theory, Analysis of Algorithms, Integral and Differential Calculus.

Course Objectives Course Objectives: Students should obtain knowledge and skills for computer solutions of mathematical problems using systems mathematical calculations.

Teaching Methods: lectures and lab exercises

Requirements/Prerequisites: Students should obtain knowledge and skills of Computer Mathematics 1, Introduction in Information Systems and Technologies, Fundamentals of Programming, Web Systems and Technologies

Assessment: two current problem tests (50%), project (20%) and computer final test exam (30%).

Registration for the Course: not necessary

Registration for the Exam: coordinated with lecturer and Student Service Department

References:

Basic Titles:

1. Anderson J. Discrete Mathematics with Combinatorics. University of South Carolina-Spartanburg, ISBN-10: 0130869988 • ISBN-13: 9780130869982, 2001, Pearson, Cloth, 799 pp, достъпна на http://www.amazon.com/Discrete-Mathematics-Combinatorics-2nd-Edition/dp/0130457914
2. Lehman Eric, F Thomson Leighton, Albert R. Meyer Mathematics for Computer Science, Massachusets Institute of Technology, 2010, електронно копие на книгата е достъпно на https://www.seas.harvard.edu/courses/cs20/MIT6_042Notes.pdf
3. Richard E. Crandall, Carl Pomerance, Prime Numbers: A Computational Perspectives, Springer Science & Business Media, Jan 1, 2001 - Computers - 545 p.
4. Ronald L. Graham, Donald E. Knuth, Concrete Mathematics. A Foundation for Computer Science, Second Edition, 1994 by Addision Wesley Publishing Company, ISBN 0-201-55802-5 (англ.), ISBN 5-03-001793-3 (руски), достъпна на http://progbook.ru/matematika/642-grehem-konkretnaya-matematika-osnovanie.html и https://notendur.hi.is/pgg/%28ebook-pdf%29%20-%20Mathematics%20-%20Concrete%20Mathematics.pdf
5. Василева М., Дискретни структури, Шумен, 2008
6. Велев Г., М. Димитров, М.Христова, Ст. Пъдевска, Висша математика в примери и задачи, УИ „Стопанство“, 2000г.
7. Генри С. Уоррен, Глава 16. Формулы для простых чисел// Алгоритмические трюки для программистов. Hacker's Delight. — М.: «Вильямс», 2007. 288 с. — ISBN 0-201-91465-4.
8. Денев, Й., С. Щраков, Дискретна математика, Благоевград, 1995.
9. С. Щраков, К. Йорджев, М. Тодорова, Ръководство за решаване на задачи по дискретна математика, Благоевград, ЮЗУ "Н. Рилски", 2005.
10. http://www.wolfram.com/mathematica/
11. http://www.math10.com/bg/algebra/visha-matematika.html
12. http://fmi.wikidot.com/anal124
13. https://sites.google.com/site/elektronniresursi/matematika

Additional Titles:

1. Д. Кнут, “Искуство программирования”, Мир, Москва, 1977.
2. Маринов М. Л. (2008) Матрично смятане с Mathematica. Издателство на НБУ, С.
3. Shingareva I., Carlos Lizarraga-Celaya, Maple and Mathematica, Springer, 2007

4. https://www.dmoz.org/Science/Math/Software
5. http://www.wolfram.com
6. http://inf-server.inf.uth.gr/~akritas/articles/Akritas_Book_Russian.pdf
7. http://www.maths.qmul.ac.uk/~pjc/notes/intalg.pdf
8. https://en.wikibooks.org/wiki/High_School_Mathematics_Extensions/Set_Theory_and_Infinite_Processes
9. http://www.cs.princeton.edu/courses/archive/spr10/cos433/mathcs.pdf
10. http://web.cecs.pdx.edu/~jhein/books/StudentStudyGuide.pdf, http://samples.jbpub.com/9780763772062/Revised_SSG.pdf

Abbreviation: FS: Fall Semester SS: Spring Semester

MATHEMATICAL LOGIC

Semester: 2
Course type: Lectures
Hours (weekly) / SS /: lectures - 2 hours per week + seminars – 1 hour per week
Number of ECTS credits: 5
Type of the course in the curriculum:
Compulsory course from the curriculum of the "Informatics" Bachelor's degree programme
Course description:
The course in mathematical logic aims to teach the basic concepts and results of propositional and predicate logic and propositional and predicate calculus. It deals with concrete first-order theories.
Goal:
The course in mathematical logic is aimed at introducing students to the development of concepts and methods of mathematical logic within the context of development in mathematics.
Teaching methods: lectures, demonstrations, problem solving
Prerequisites: The acquired knowledge is useful.
Examination and assessment procedures: The estimation of the acquired knowledge is based on a written exam which consists of problem solving and theoretical knowledge examination (writing on a topic from the syllabus provided to students)
Course enrolment: Students should submit an application at the academic affairs department at the end of the current semester
Registration for examination: coordinated with the lecturer and the academic affairs department
References:

Basic

1. Въведение в математическю логику, Е. Менделсон
2. Сказки по логика, С. Паси и колектив
3. Приицесса или тигр?, Р. Смаллиан, "Мир", Москва 1985.

Additional

1. A concept of logic, Seventh edition. Hurley, Springer, 2009, http://ihtik.lib.ru/2012.03_ihtik_mathematic/

2. Set Theory and Logic, Robert Roth Stoll, Springer 2009.

3. Applied Computer Science, Shane Torbert, 2011.

4. Concise Guide to Computation Theory, Akira Maruoka, 2011.

5. How to Solve It: A New Aspect of Mathematical Method, George Pólya, 2008.

Notes:
SS: Spring semester

OBJECT-ORIENTED PROGRAMMING

Title: Object-Oriented Programming

Semester: 2 semester

Type of Course: lectures, seminars and labs

Hours total – 30 lectures + 30 labs

Credits Numbers: 8,0

Department: Informatics, Tel.: +359 73 8889 132

Course Status: Fundamental course from the Computer Science Bachelor Curriculum.

Course description: In the course students are introduced with methods and means of Object-oriented programming The course is providing basic knowledge in development of algorithms, their programming using particular programming language and running and testing of the programs under certain operation system. The structure and the main operational principles of the computer systems are given. The means and accuracy of information presentation are also considered. Some of the key classes of algorithms and data structures are studied. The main techniques of the structural approach of programming and their application using C++ programming language are introduced. The aim of the course is to teach the students with the techniques in development of algorithms and programs using C++ programming language. The knowledge will be used in the general theoretical, technical and some special courses.

Objectives:

Basic objectives and tasks:

- The students give knowledge for algorithum thinking;
- to give knowledge for methods and skils in Object-oriented programming in integrated development environment for visual programming;
- to give knowledge for Data structures, that can process with computer;
- To give knowledge for methods and skills in programming.
- to give knowledge for good style in programming;

- to give knowledge for basic principles when develop applications

Methods of teaching: lectures, tutorials, group seminars or workshop, projects, other methods

Pre-requirements: The course is continued of the course "Introduction in programming".

Basic knowledge in Mathematic.

Exam: Written examination and discussion at the end of the semester, individual tasks and the general student's work during the semester.

Registration for the Course: not necessary

Registration for the Exam: Coordinated with the lecturer and the Student Service Office

References:

А). Основна литература:

1. Павел Азълов, *Обектно-ориентирано програмиране. Структури от данни и STL.*София, Сиела, 2008.
2. Магдалина Тодорова, *Програмиране на C++.* Сиела, 2002.
3. М. Тодорова*, Обектно-ориентирано програмиране на базата на езика C++.* София, Сиела, 2011.
4. Кай Хорстман, *Принципи на програмирането със C++.* Софтех, 2000.

Б) Допълнителна литература:

5. Хърбърд Шилдт, *C++ Практически самоучител.* Софтпрес, 2001.
6. Стийв Донован, *C++ в примери.* Софтпрес, 2005.
7. Димитър Богданов, *Обектно-ориентирано програмиране със C++.* София, Техника, 2002.
8. Христо Крушков, *Програмиране на C++.* Пловдив, Макрос, 2006.
9. Брайън Овърленд, *C++ на разбираем език.* АлексСофт, 2003.
10. Магдалина Тодорова, Петър Армянов, Дафина Петкова, Калин Георгиев, Сборник от задачи по програмиране на C++. Първа част – Увод в програмирането. София, ТехноЛогика, 2008.
11. Магдалина Тодорова, Петър Армянов, Калин Георгиев, Сборник от задачи по програмиране на C++. Втора част – Обектно-ориентирано програмиране. София, ТехноЛогика, 2008.
12. Преслав Наков, Панайот Добриков *Програмиране, = ++ алгоритми.* София, 2005.
13. К. Ш. Тан, В.-Х. Стиб, Й. Харди, *Символны C++. Введение в компьютерную алгебру с использованием объектно-ориентированного программирования.* Москва, Мир, 2001.
14. Г. С. Иванова, Т. Н. Ничушкина, Е. К. Пугачев, *Объектно-ориентированное программирование.* Москва, МГТУ, 3003.

ENGLISH LANGUAGE 2

Semester: 2
Course type: Seminars
Hours (weekly) / WS / SS: 2 hours per week/SS

Number of ECTS credits: 2, 5

Type of the course in the curriculum:

Compulsory course from the curriculum of the "Informatics" Bachelor's degree programme
Course description:

Introducing students to the basic components of English phonology, morphology and syntax. It helps students learn and practice communicating in everyday situations including asking and answering questions, using the telephone, taking messages, initiating conversations, asking for directions, making invitations and closing conversations. Class activities include role-playing, small-group activities and short presentations. It also develops skills in reading speed and comprehension. Students are introduced to reading strategies such as skimming, scanning, guessing meaning from context, previewing, predicting, making inferences and giving opinions. Reading materials include short stories, news articles, computer passages and a simplified novel.

Goal:
The goals of the course is to enable students to speak and write effectively and confidently in their professional and personal lives. Students become acquainted with the basic terminology in the specific field.

Teaching methods: Seminars
Prerequisites: The knowledge acquired at high school is useful.
Examination and assessment procedures: The estimation of the acquired knowledge is based on a written exam

Course enrolment: students should submit an application at the academic affairs department at the end of the current semester
Registration for examination: coordinated with the lecturer and the academic affairs department
References:

7. English for Mathematicians and Computer Scientists, Sofia 2001
8. Raymond Murphy, English Grammar in Use I, Prosveta, Sofia, 1994
9. Raymond Murphy, English Grammar in Use II, Prosveta, Sofia, 1994
10. Soars, John & Liz, New Headway Elementary, Oxford University Press, 2000
11. Soars, John & Liz, New Headway Pre-Intermediate, Oxford University Press, 2007
12. Ранкова, М., Иванова, Ц., Английска граматика, Наука и изкуство, София, 1998

Notes:
SS: Spring semester

DISCRETE MATHEMATICS

Semester: 3

Course Type: lectures/tutorials/

Hours (weekly): 4+4/SS

ECTS Credits: 9,0

Course Status: Compulsory/ BSc/

Short Description: The course includes the following topics: Set theory, Graph theory, Finite state automata, Formal languages, Turing's machine, Decidability, Discrete functions.

Course Aims: The aim of the course is to provide knowledge on the theoretical basis of the computing, on the formal languages, on discrete structures etc.

Teaching Methods: The knowledge from the lectures are used in the tutorials to solve theoretical and practical problems concerning the course topics.

Requirements/Prerequisites: minimum knowledge about matrices, graphs, finite algebraic systems, number theory

Exam: final exam

Registration for the course: A request is made by students at the end of the current semester

Registration for the exam: Coordinated with lecturer and Students Service Department

References:

1. Денев, Й., С. Щраков, *Дискретна математика*, Благоевград,1995

2. Павлов, Р., С. Радев, С. Щраков, *Математически основи на информатиката*,

Благоевград, 1997

3. Денев, Й., Р. Павлов, Я. Деметрович, *Дискретна математика*, София, 1984

4. Т.Фудзисава, Т.Касами, *Математика для радиоинжинеров*, Радио и связь, Москва, 1984.

5. К.Чимев, Сл.Щраков, *Математика с информатика*, Благоевград, 1989.

6. С.В.Яблонски, Введение *в дискретную математику*, М.,1979.

7. С.В.Яблонски, Г.П.Гаврилов, В.Б.Кудрявцев, *Функции алгебры логики и классы Поста*, М.,1966.

8. Z.Manna, *Mathematical theory of computation*, McGraw-Hill Book Company, NY, 1974.

9. V.J.Rayward-Smith, A *first course in formal language theory*, Bl.Sc.Publ. London, 1983.

10. A.Salomaa, Jewels *of formal language theory*, Comp.Sc.Press, Rockville, 1981.

Abbreviation:

FS: Fall Semester, SS: Spring Semester

PROGRAMMING AND DATA STRUCTURES

Title: Programming and data structures

Term: 3th term

Type of Course: lectures, seminars and labs

Hours – 45 lectures + 15 seminars + 15 labs

Credits Numbers: 5.5

Department: Informatics, Tel.: +359 73 8889 132

Course Status: Fundamental course from the Computer Science Bachelor Curriculum.

Course description: Programming and data structures is a basic course for second year students (third term) in the curriculum of Informatics. The main goal of the course is to master the tools and methods of the C ++ language, related to the modeling of different types of data structures. Basic algorithms are considered - sorting and searching, and recursion as a programming method. Linear data structures and binary trees are studied. Basic techniques of C ++ programming are also presented, such as dividing the program into modules, compiling separately and organizing projects. It is a natural continuation of the Object-Oriented Programming course, which is studied in the second term. The acquired knowledge is also used by other courses in informatics, programming and databases. Students must have taken the following subjects: Introduction to Programming and Object Oriented Programming. As a basic course in Computer science, it is necessary for the fuller mastery of many other disciplines included in the curriculum.

Objectives:

Basic objectives and tasks:

- The students give knowledge for algorithm thinking;
- to give knowledge for Data structures, that can process with computer;
- to give knowledge for methods and skills in programming.
- to give knowledge for good style in programming;
- to give knowledge for basic principles when develop applications

Methods of teaching: lectures, tutorials, group seminars or workshop, projects, other methods

Pre-requirements: Basic knowledge in Programing and OOP.

Exam: Written examination at the end of the term.

Registration for the Course: not necessary

Registration for the Exam: Coordinated with the lecturer and the Student Service Office

References:

1. Кай Хорстман, Принципи на програмирането със C++, ИК Софтех, София, 2000.

2. Николай Киров, Сборник от учебни материали по Програмиране и структури от данни, Деметра, София, 2004.
3. Магдалина Тодорова, Програмиране на C++ (част втора), Сиела, София, 2002.
4. Лендерт Амерал, Алгоритми и структури от данни в C++, ИК "Софтех", 2001.
5. Бьорн Строустроп, Програмният език C++. Специално издание. Том 1 и 2, ИК "Инфодар", София, 2001.
6. Иван Плачков Ръководство по програмни езици. УниСофт-Пловдив, 2000.
7. Michael Goodrich, Roberto Tamassia, David M. Mount, Data Structures and Algorithms in C++, Wiley, 2004.
8. Michael Goodrich, Roberto Tamassia, David M. Mount, Data structures, and Algorithms in C++, Wiley, Second edition, 2011.

DIFFERENTIAL EQUATIONS AND APPLICATIONS

Semester: 3

Course Type: lectures and seminars

Hours (weekly): 2 / 2 (FS)

ECTS Credits: 5,5

Course Status: Compulsory course from Computer Science B.C. Curriculum

Course Description:

Mathematical methods of investigation are used in every field of science and technology.

Differential Equations are the foundations of the mathematical education of scientists and engineers. The main topics are: First-order Linear equations with constant coefficients: exponential growth, comparison with discrete equations, series solutions; modeling examples including radioactive decay and time delay equation. Linear equations with non-constant coefficients: solution by integrating factor, series solution. Nonlinear equation: separable equations, families of solutions, isoclines, the idea of a flow and connection vector fields, stability, phase-plane analysis; examples, including logistic equation and chemical kinetics. Higher-order Linear equations: complementary function and particular integral, linear independence, reduction of order, resonance, coupled first order systems. Examples and PC-models of nonlinear dynamics, order and chaos, attractors, etc.

Course Aims:

The main goal is the students to master the instruments and methods of modelling in science.

Teaching Methods: lectures, tutorials, homeworks, tests, etc.

Requirements/Prerequisites: Calculus I and II, Linear Algebra and Analytical Geometry.

Exam: tests, homeworks, final exam

Registration for the course: compulsory course

Registration for the exam: Coordinated with lecturer and Students Service Department

References:

1. Popivanov P, PKitanov. Ordinary Differential equations. Blagoevgrad, 2000 (in Bulgarian).

2. Baynovq D, N. Kazakova Ordinary Differential equations. PU,1987. (in Bulgarian).

3. Elsgolc. L. Differential equations and variational calculus, Moscow, 2000 (in Russian)

4 . Pontryagin L.S., Differential equations and applications, Moscow, 1988 (in Russian).

5. Boss. Lectures in mathematics. Differential equations, Moscow, 2004 (in Russian).

6 Stewart J. . Calculus. III ed. (AUBG). 1996.

7. D. Pushkarov, Mathematical methods in physics, Part I, Blagoevgrad, 1993 (in Bulgarian).

8 F. Diacu. An Introduction to Differential Equations: Order and Chaos, Freeman, 2000, etc.

Abbreviation:

FS: Fall Semester

COMPUTER ARCHITECTURES

Semester: **3**

Form of the course: **Lectures/seminars**

Hours (per week): **3 hours lectures + 1 hours exercises per week, fall semester**

Credits: **5 credits**

Status of the course in the educational plan:

The course is compulsory in BSc curriculum in Informatics.

Description of the course:

The course covers the advanced computer systems, their programming and functional model, introduce information in computer organization and memory types (major, operational, permanent, outdoor, etc.), system interruptions, features and technology solutions, conveyor ADP modes, system bus (types and structures), some problems of modern computer architectures (RISC, parallel and multiprocessor computer systems).

Scope of the course:

Obtaining knowledge of a systematic overview of the modern computer architecture, systems to form the theoretical and practical basis for better understanding of the work of computers to acquire skills in programming in assembly language.

Methods: discussions, practical exercises of the codes under consideration

Preliminary requirements: The students must have basic knowledge from mathematics.

Evaluation: permanent control during the semester (two written exams) and final exam.

Registration for the course: by application in the Educational Office

Registration for exam: up to agreement with the teacher and the Educational Office

Literature:

1. Брадли, Д. "Програмиране на асемблер за персонален компютър IBM/PC" Техника, София, 1989

2. Иванов Р. "Архитектура и системно програмиране за Pentium базирани компютри", Габрово, 1998.

3. J. L. Hennessy, D. A. Patterson. Computer Architecture: A Quantitative Approach (3rd ed.). Morgan Kaufmann Publishers, 1996.

4. Боровски Б, Боровска П, Архитектура на ЕИМ и микрокомпютри, Техника, 1992.

5. Горслайн Дж, Фамилия ИНТЕЛ, Техника, 1990.

6. Въчовски И, Наръчник по 32-разредни микропроцесори.

7. Компютърна енциклопедия, издателство Nisoft, част I и II.

NUMBER THEORY

Semester: **3**

Form of the course: **lectures**

Hours (per week): **3 hours lectures**

Credits: **5 credits**

Status of the course in the educational plan:

The course is optional in BSc curriculum of Informatics.

Description of the course:

The course starts with introduction of the foundation of the number theory: divisibility, initial theory of the congruences, the main theorem of the Arithmetic, the theorems of Fermat and Euler. The structure of the group of remainders which are relatively prime with the module is considered. The theory of the congruences of one and several unknowns is developed. The quadratic residues are introduced and the criterion of Euler is proved. Some Diophantine equations and the function [x] are considered.

Scope of the course:

Obtaining knowledge of the theoretical backgrounds and practical abilities for applications of the Number theory.

Methods: lectures, discussions on the methods of solving congruences, other problems from the Number theory.

Preliminary requirements: The students must have basic knowledge from the high school courses and elementary notions of the group theory.

Evaluation: permanent control during the semester (two written exams) and exam in the semesters end in two parts – problems solving and answering theoretical questions.

Registration for the course: by application in the Educational Office in the end of the semester

Registration for exam: up to agreement with the teacher and the Educational Office

Literature:

1. Notices (www.moi.math.bas.bg/~peter)
2. Додунеков Ст., К. Чакърян. Задачи по теория на числата, Регалия, София

MATHEMATICAL ANALYSIS 3

Semester: **3**

Course Type: l**ectures/tutorials**

Hours (per week): **lectures - 2 hours/ tutorials- 1 hour per week**

ECTS credits**: 5 credits**

Course Status: Optional for students in BSc in Informatics.

Description of the course:

Main topics to be considered:

- Implicit function theorem, conditional extrema
- Line integrals
- Multiple integrals
- Fourier series
- Complex numbers
- Holomorphic functions and Power series
- Elementary transcendental functions
- Integrals of complex-valued functions, Cauchy theorem and applications
- Taylor and Laurant series
- Classification of isolated singular points
- Residue and applications

Course Aims: This course is an introduction to the classical Complex Analysis and its main purpose is to present some basic topics of the Theory of holomorphic functions of one variable as well as some of its applications

Teaching Methods: lectures, tutorials and homework projects

Requirements/Prerequisites: Mathematical Analysis, Parts I and II, Analytic Geometry

Assessment: Written final exam

References:

1. Дойчинов, Д. Математически анализ в крайномерни пространства, Наука и изкуство, София, 1979

2. Любенова Е., Недевски П., Николов К., Николов Л., Попов В., - Ръководство по математически анализ 1 част, УИ "Кл. Охридски"

3. Аргиров Т. Теория на аналитичните функции, Наука и изкуство, София, 1988

4. Чакалов Л. Увод в теорията на аналитичните функции. Наука и изкуство, София, 1957

5. Алфорс Л. Увод в теорията на аналитичните функции. Наука и изкуство, София, 1971 /превод от английски/

6. Маркушевич А.И. Маркушевич Л. И. Въведение в теорию аналитических функции. М., 1977

7. Аргирова Т., Генчев Т. Сборник от задачи по теория на аналитичните функции. Наука и изкуство, София, 1986 /трето издание/

8. Аргирова Т., Генчев Т. Дробно-линейна функция. Наука и изкуство, София, 1971.

DISCRETE FUNCTIONS

Semester: 3

Course Type: lectures and tutorials

Hours per Week/FS/SS: 2 lecture hours and 1 tutorial hour per week/ FS

ECTS Credits**: 5 credits**

Course Status:

Optional course in Computer Science B.S. Curriculum

Course Description: The following basic topics are studied:

- functional constructions in κ-valued logic

- essential and strong essential dependence of functions of κ-valued logic upon their arguments

- complexity of realization of boolean functions

- logic methods for analysis and synthesis of schemes.

Teaching Methods:

During lecture hours, key problems are discussed. This discussion is detailed during seminar hours. Students report what they have studied as well as results obtained by them via independent work.

Requirements/Prerequisites: **Mathematical logic.**

Assessment: essay/paper or written exam and oral exam. The grade also depends on the students' work during the semester.

Registration for the Course: by request at the end of the current semester

Registration for the Exam: coordinated by lecturer and Student Service Department

References:

1. Yablonskii S.V., Functional Constructions in κ-valued logic. Proceedings of V.A. Steklov Institute of Mathematics, 51, 1958, 5-142. (in Russian)

2. Chimev, K., Functions and Graphs. Blagoevgrad, 1984, 1-199. (in Bulgarian)

3. Chimev, K., I. Giudzhenov. Subfunctions and cardinality of some classes of functions. Blagoevgrad, 1991, 1-220. (in Bulgarian)

4. Chimev, K., Discrete functions and subfunctions, Blagoevgrad, 1991, 1-258. (in Bulgarian)

5. Chimev, K., Iv. Mirchev. Separable and dominating sets of variables, Blagoevgrad, 1993, 1-131.

6. Shtrakov, S. Dominating and annuling sets of variables for the functions. Blagoevgrad, 1987, 1-180.

Abbreviations:

FS: Fall Semester

SS: Spring Semester

SPECIAL MATRICES

Semester: 3

Course Type: lectures and tutorials

Hours per Week/FS/SS: 2 lecture hours and 1 tutorial hour per week/ FS

ECTS Credits**: 5 credits**

Course Status: Optional course in Computer Science B.S. Curriculum

Course Description: The optional course in Special Matrices has the objective to make the students familiar with: the basic sort of matrices, which have place in different fields of Mathematics and their applications; methods for their reduction to canonical form; some unsolved problems in the field of the matrices like Hadamard matrices. The course extends and expands the students' knowledge, which they have from the course in Linear Algebra. A great part of this material is an introduction into mathematical speciality at the Universities, which make efforts to use modern methods for research with the help of Mathematics and Computer Sciences. The lecture course shows the current status of this material. This course is built, based on the notation of "vector space".

Course Aims: Students should obtain knowledge and skills for basic concepts under consideration. They should successfully use the methods for calculating them.

Teaching Methods: lectures and tutorials.

Requirements/Prerequisites: Linear Algebra, Analytic Geometry.

Assessment: written final exam.

Registration for the Course: by request at the end of the current semester

Registration for the Exam: coordinated by lecturer and Student Service Department

References:

1. YKl. Denecke, K. Todorov. "Linear Algebra", Neofit Rilski University Press, Blagoevgrad, 1992. (in Bulgarian)

2. K. Dochev, D. Dimitrov. "Linear Algebra", Science and Art, Sofia, 1973. (in Bulgarian)

3. A. Hedayat, W.D. Wallis. Hadamard Matrices and Their Applications, Annals of Statistics, 6 (1978) No. 6, pp. 1184-1238.

Abbreviations:

FS: Fall Semester

SS: Spring Semester

INTRODUCTION IN INFORMATION SYSTEMS AND TECHNOLOGIES

Semester: 3

Type of Course: Lectures and tutorials in computer lab.

Hours per week: 2 hours lectures and 1 hours tutorials in computer lab/autumn semester.

Credits Numbers: 5 credits

Course Status: Optional course in Computer Science B.S. Curriculum.

Course description: The course involves basic concepts as information, data, knowledge, information system, business information systems, hardware and software components od IS ets. The problems related to ICT jobs, copyrights and law issues in ICT.

Objectives: The student should obtain basic knowledge in area of IT and IS:

Methods of teaching: lectures, tutorials, discussions, project based method.

Pre - requirements: No (Introductory course)

Assessment and Evaluation:

Project- 50%

Final Test- 50%

The course is successful completed with at least 65% of all scores.

Registration for the Course: not required (core course)

Registration for the Exam: coordinated with the lecturer and the Student Service Office

References:

1. Ralph M. Stair, George W. Reynolds, Fundamentals of Information Systems, Sixth Edition, 2012 Course Technology, Cengage Learning
2. BRIAN K. WILLIAMS, I. STACEY C. SAWYER, Using Information Technology. A Practical Introduction to Computers & Communications, McGraw-Hill, 2011

URL http://www.e-learning.swu.bg

DEVELOPMENT OF OBJECT-ORIENTED APPLICATIONS WITH DESIGN PATTERNS

Semester: 3

Course Type: lectures and labs

Hours (weekly)/WS/SS: 2 lectures and 1 lab per week / WS

ECTS Credits: 5 credits

Course Status: Optional course.

During the course students will become familiar with different types of design patterns and opportunities for code reuse in development of object-oriented applications. Following classical taxonomy regarded as object-oriented patterns (GoF design patterns), and architectural patterns will be observed. GoF patterns are grouped into three main groups - building, structural and behavioral. The course addresses some of the so-called. anti-patterns. For each one of them a rigorous analysis of its weaknesses will be performed.

Course Objectives: The course aims to broaden the usual preparation of programmers with specialized training in the application of design patterns.

Teaching Methods: Lectures, demonstrations, work on project and teamwork.

Requirements/Prerequisites: Needed basic knowledge of operating systems, programming, computer architectures, computer networks and communications.

Assessment: written final exam

Course enrolment: Students should submit an application at the academic affairs department at the end of the current semester

References:

1. Erich Gamma, Richard Helm, Ralph Johnson and John Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley, 1994
2. Eddie Burris, Programming in the Large with Design Patterns, Pretty Print Press, 2012
3. Tony Bevis, C# Design Pattern Essentials, Ability First Limited, 2012
4. Martin Fowler, Patterns of Enterprise Application Architecture, Addison-Wesley,

2003

5. Cay Horstmann, Object-Oriented Design and Patterns, Wiley, 2006
6. Eric Freeman, Bert Bates, Kathy Sierra and Elisabeth Robson, Head First Design Patterns, O'Reilly, 2004

Tony Bevis, Java Design Pattern Essentials - Second Edition, Ability First Limited, 2012

SOFTWARE QUALITY ASSURANCE

Semester: 3

Course Type: lectures

Hours (weekly)/WS/SS: 1 lecture and 2 labs per week / SS

ECTS Credits: 5,0 credits

Course Status: Optional Course.

The course consider the role of QA in the process of software development. The course covers basic ideas, views and major trends on the concept of software quality regarding to standards. Various QA methods such as White Boxes, Black Boxes, Gray Boxes are introduced as well as principles, stages and types of software testing. Various tools for automated testing and bug tracking platforms are studied. Some metrics statistical and probabilistic methods and approaches for assessing the quality of the software product are introduced.

Course Objectives: The course aims to expand the training of students majoring in "CST" in the field of quality control software.

Teaching Methods: Lectures, demonstrations, work on project and teamwork.

Requirements/Prerequisites: Needed basic knowledge of operating systems, programming, computer architectures, computer networks and communications.

Assessment: written final exam

Course enrolment: the course is compulsory

References:

1. Иванов М.П., И. Момчев, Принципи и проблеми на многокритериалната оценка на качеството на софтуерния продукт, сп."Автоматика и информатика", 2006
2. Мартин Иванов, Принципи и перспективи за оценка на качеството на софтуерните продукти (http://eprints.nbu.bg/505/1/PRINCIPLES_AND_PERSPECTIVES.pdf)
3. ISO, International Organization for Standardization, "ISO 9126-1:2001, Software engineering – Product quality, Part 1: Quality model", 2001.
4. Stephen Kan H. , Metrics and Models in Software Quality Engineering, 2nd Edition, AddisonWesley Professional., 2002.
5. J. Kuruvilla, JIRA 5.x Development Cookbook, Packt Publishing, 2013

6. Elfriede Dustin, Jeff Rashka, John Paul, Automated Software Testing: Introduction, Management, and Performance, Addison-Wesley Professional, 1999
7. James D. McCaffrey, Software Testing: Fundamental Principles and Essential Knowledge, 2009
8. Patrik Berander, Software quality attribute and trade-offs, Editors: Lars Lundberg, Michael
9. Mattsson, Claes Wohlin, Blekinge Institute of Technology , June 2005
10. Dick S., A. Kandel, Computational Intelligence In Software Quality Assurance, Series in Machine Perception and Artificial Intelligence - Vol. 63, 2005.

INTRODUCTION TO XML

Semester: 3

Course Type: lectures and laboratory exercises

Hours per week FS/SS: 1 lecture hour and 2 laboratory hours per week /FS

ECTS credits: 5 credits

Course Status: Optional Course.

Course Description: The course includes basics of the language XML and includes topics: introduction to the language XML, advantages of presenting data through XML; well-formed XML documents; use of namespaces in XML; validation of XML documents using DTD; validation of XML documents through schemes; create XML schemas; XML languages schemes Relax NG and Schematron; extracting data from XML documents, use DOM; view language xPath; getting to know the language XSLT, key elements; reuse of code in XSLT, built-In templates and rules; getting to know the language xQuery; additional features of the language xQuery; XML and databases; employment opportunities for the management of databases with XML;

Course Objectives: Students should obtain fundamental knowledge and skills related to the basics of the language XML and related technologies.

Teaching Methods: lectures and laboratory exercises

Requirements/Prerequisites: Basic knowledge and skills for information systems and technology.

Assessment: written final exam

Registration for the Course: by request at the end of the previous academic year

Registration for the Exam: coordinated with lecturer and Student Service Department

References:

1. Joe Fawcett, Liam R.E. Quin, Danny Ayers. Beginning XML, Fifth Edition. John Wiley & Sons, Inc. 2012.
2. Elliotte Rusty Harold. XML 1.1 Bible, 3rd Edition. Wiley Publishing, Inc. 2004.

3. Dorothy J. Hoskins. XML and InDesign, First Edition. O'Reilly Media, Inc. 2013.
4. Aaron Skonnard, Martin Gudgin. Essential XML Quick Reference: a Programmer's Reference to XML, XPath, XSLT, XML Schema, SOAP, and More. Pearson Education, Inc. 2002.
5. Doug Tidwell. XSLT, Second Edition. O'Reilly Media, Inc. 2008.
6. Priscilla Walmsley. XQuery, First Edition. Priscilla Walmsley. 2007.

Abbreviation:

FS: Fall Semester

SS: Spring Semester

MATHEMATICAL OPTIMIZATION

Semester: 4

Course Type: lectures and tutorials

Hours per week/FS/SS: 3 lecture hours and 2 tutorial hours per week/SS

ECTS credits: 8 credits

Course Status: Compulsory course in the Informatics Science B.S. Curriculum

Short Description:

The course in Optimization (Mathematical Programming) includes basic results and methods for solving various types optimization problems and related topics: nonlinear optimization problems, linear optimization problems (simplex method, duality in linear optimization, transportation problem, assignment problem), matrix games (John von Neumann minimax theorem, graphical method for solving 2 x 2, 2 x *n,* and *m* x 2 games, relation between matrix games and linear optimization), convex analysis (convex sets, sum of sets and product of a set with a real number, projection of a point onto a set, separation of convex sets, extreme points, cones, polar cones, representation of convex cones, representation of convex sets, polyhedrons, convex functions, directional derivatives, subgradients and subdifferentials), convex optimization problems (Kuhn-Tucker theorem), quadratic optimization problems.

Course Aims:

Students should obtain basic knowledge and skills for solving optimization problems under consideration.

Teaching Methods: lectures and tutorials

Requirements/Prerequisites: Mathematical Analysis, Linear Algebra, Analytic Geometry.

Assessment: written final exam

Registration for the Course: not necessary

Registration for the Exam: coordinated with the lecturer and Students Service Department

References:

Basic Titles:

1. P. Kenderov, G. Hristov, A. Dontchev – "Mathematical Optimization", Kliment Ohridski Sofia University Press, Sofia, 1989 (in Bulgarian).
2. "Mathematical Optimization Problem Book and Handbook", Kliment Ohridski Sofia University Press, Sofia, 1989 (in Bulgarian).
3. M. Slavkova – "Mathematical Optimization Methods", Sofia, 2000 (in Bulgarian).
4. M. Slavkova, Z. Tsenova – "Quantitative Methods and Statistics Problem Book", Technical University, Sofia, 2011 (in Bulgarian).
5. S. M. Stefanov – "Quantitative Methods of Management", 2003 (in Bulgarian).

Additional Titles:

6. Suresh Chandra, Jayadeva Aparna Mehra – "Numerical Optimization with Applications", Narosa Publishing House, New Delhi, 2013.
7. Andrew R. Conn, Katya Scheinberg, Luis N. Vicente – "Introduction to Derivative-Free Optimization", SIAM, Philadelphia, PA, USA, 2009.
8. I. Griva, S. G. Nash. A. Sofer – "Linear and Nonlinear Optimization", 2-nd. ed., SIAM, Philadelphia, 2009.
9. S. M. Stefanov – "Separable Programming. Theory and Methods", 4-th ed., Springer, Dordrecht-Boston-London, 2016.
10. Hamdy A. Taha – „Operations Research. An Introduction", 10-th ed., Pearson, USA, 2017.
11. William F. Trench – "The Method of Lagrange Multipliers", Trinity University, San Antonio, Texas, USA, 2013 (also available online).

Abbreviation: FS: Fall Semester **SS**: Spring Semester

OPERATING SYSTEMS

Semester: **4**

Type of Course: **lectures and tutorials in computer lab**

Hours per week: **2 hours lecture and 3 hours tutorials in computer lab**

Credits Numbers: **7,0 credits**

Course Status: Basic course in curriculum of major Informatics. Bachelor degree.

Course description:

A theoretical side including; Operating system structure and services, processor scheduling, concurrent processes, memory management, virtual memory, input/output, secondary storage management, and file systems. Also an applicable side include; Installing, partitioning, configuring and upgrading Windows. Troubleshooting (Common errors and problems and how to solve them). Identifying the networking capabilities of Windows including procedures for connecting to the network.

Objectives:

- Recognize the concepts and principles of operating systems.
- The main objective of this course is to provide students with the basic knowledge and skills of operating, managing, and maintaining microcomputer systems.
- Hands-on experience with the MS Windows environment will be a major concern in this course.

Methods of teaching: lectures, tutorials, discussions, project based method.

Pre - requirements: Database systems (core course)

Assessment and Evaluation

Quizzes - 30%

Final Test - 70%

The course is successful completed with at least 65% of all scores.

Registration for the Course: No (core course)

Registration for the Exam: coordinated with the lecturer and the Student Service Office

References

1. Лилян Николов, Операционни системи, ИК "Сиела", София, 1998.

2. William Stallings, Operating Systems: Internals and Design Principles, Prentice Hall, Third Edition 1998; Fifth Edition 2005.

DATA ANALYSIS WITH MS EXCEL AND VBA

Semester: 4

Type of Course: lectures and tutorials in computer lab.

Hours per week: 1 hours lectures and 2 hours tutorials in computer lab/ spring semester.

ECTS Credits: 4 credits

Course Status: Optional course.

Course description: The course is an introduction to data analysis with MS Excel and VBA.

Objectives:

The student should obtain knowledge of:

- Design and implementation of VBA applications.
- Design and implementation data analysis using MS Excel and VBA.

Methods of teaching: lectures, tutorials, discussions, project based method.

Pre- requirements: Database systems (core course).

Assessment and Evaluation

Project- 50%

Final Test- 50%

The course is successful completed with at least 51% of all scores.

Registration for the Course: by request at the end of the current semester

Registration for the Exam: coordinated with the lecturer and the Student Service Office

References:

Basic

1. Пол Макфедрис, Excel 2016 Формули и функции, ЗеСТ Прес, 2017
2. J. Parsons, D. Oja, R. Ageloff, P. Carey, New Perspectives on Microsoft Excel 2016, Comprehensive, Course Technology Cengage Learning, 2016

Additional

1. Walkenbach, John, Excel® 2013 Power Programming with VBA, John Wiley & Sons, Inc, 2013
2. https://www.guru99.com/creating-your-first-visual-basic-for-applications-vba-in-excel.html (2020)

On-line resources

3. URL http://www.e-learning.swu.bg

PROGRAMMING WITH .NET FRAMEWORK

Semester: 4

Course Type: lectures and labs

Hours (weekly)/WS/SS: 1 lecture and 2 labs per week / SS

ECTS Credits: 4 credits

Course Status: Optional course.

Short Description: This course observes .NET software development framework. The main topics included in this course are: .Net Framework overall architecture, CLR, CTS, lambda expressions, data access – EF/LINQ, processing XML, WinForms. Will be addressed and some of the more complex concepts such as reflection, asynchronous programming.

Course Aims: The course aim is to give theoretical and practical background to students to use .NET family languages in custom software development.

Teaching Methods: Lectures, Labs, Discussions, Project Based Methods

Requirements/Prerequisites: Knowledge in Programming Basics, Object Oriented Programming with C++/Java, Operating Systems.

Exam: final exam

Course enrolment: Students should submit an application at the academic affairs department at the end of the current semester

Registration for the exam: Coordinated with lecturer and Students Service Department

References:

1. Светлин Наков и Веселин Колев, Въведение в програмирането със C#, Фабер, 2011, ISBN: 978-954-400-527-6
2. Eric Gunnerson and Nick Wienholt, A programmer's Guide to C# 5.0, APress, 2012
3. Daniel Solis, Illustrated C# 2012, 2nd.Edition. APress, 2012
4. Thuan Thai and Hoang Lam, .NET Framework Essentials, 2nd Edition, O`Reilly, 2002, ISBN 0-596-00302-1
5. Jeff Prosise, Programming Microsoft .NET (core reference), Microsoft Press, 2002, ISBN 0-7356-1376-1
6. Jesse Liberty, Programming C#, 2nd Edition, O`Reilly, 2001, ISBN 0-596-00117-7
7. Fergal Grimes, Microsoft .NET for Programmers, Manning Publications, 2002, ISBN 1-930110-19-7
8. Jesse Liberty, Programming C#, 2nd Edition, O'Reilly & Associates, Inc., 2002, 648 pages, ISBN: 0596003099

Microsoft Developers Network. https://msdn.microsoft.com/bg-bg

GRAPH THEORY

Semester: 4

Cours Tipe: Lectures and tutorials

Hours per week/FS/SS: 2 lecture hours and 1 tutorial hour

ECTS credits: 4.0 credits

Course Status: Optional course in the Computer Science B.S. Curriculum

Short Description:

The 1970s ushered in an exciting era of research and applications of networks and graphs in operations research, industrial engineering, and related disciplines. Graphs are met with everywhere under different names: "structures", "road maps" in civil engineering; "networks" in electrical engineering; "sociograms", "communication structures" and "organizational structures" in sociology and economics; "molecular structure" in chemistry; gas or electricity "distribution networks" and so on.

Because of its wide applicability, the study of graph theory has been expanding at a very rapid rate during recent years; a major factor in this growth being the development of large and fast computing machines. The direct and detailed representation of practical systems, such as distribution or telecommunication networks, leads to graphs of large size whose successful analysis depends as much on the existence of "good" algorithms as on the availability of fast computers. In view of this, the present course concentrates on the development and exposition of algorithms for the analysis of graphs, although frequent mention of application areas is made in order to keep the text as closely related to practical

problem-solving as possible. Although, in general, algorithmic efficiency is considered of prime importance, the present course is not meant to be a course of efficient algorithms. Often a method is discussed because of its close relation to (or derivation from) previously introduced concepts. The overriding consideration is to leave the student with as coherent a body of knowledge with regard to graph analysis algorithms, as possible.

In this course are considered some elements of the following main topics:

- Introduction in graph theory (essential concepts and definitions, modeling with graphs and networks, data structures for networks and graphs, computational complexity, heuristics).
- Matching and assignment algorithms (introduction and examples, maximum-cardinality matching in a bipartite graph).
- The chinesse postman and related arc routing problems (Euler tours and Hamiltonian tours, the postman problem for undirected graphs, the postman problem for directed graphs).
- The traveling salesman and related vertex routing problems (Hamiltonian tours, basic properties of the traveling salesman problem, lower bounds, optimal solution techniques, heuristic algorithms for the TSP).

Course Aims:

Students should obtain basic knowledge in Graph theory and skills for solving optimization problems for graphs and networks.

Teaching Methods: lectures, tutorials, individual student's work

Requirements/Prerequisites: Graphs, Discretion Programming

Assessment: 3 homework D1, D2, D3; 2 tests K1, K2 (project); written final exam

Rating: $= 0{,}2 .(\frac{\mathbf{D}1+\mathbf{D}2+\mathbf{D}3}{3}) + 0{,}5 .(\frac{\mathbf{K}1+\mathbf{K}2}{2}) + 0{,}3$ (Exam)

Registration for the Course: by request at the end of the current semester.

Registration for the Exam: coordinated with the lecturer and Students Service Department

References:

1. Mirchev,Iv., "Graphs". "Optimization algorithms for networks", Blagoevgrad, 2001 (in Bulgarian).

2. Mirchev,Iv., "Mathematical Programming", Blagoevgrad, 2000 (in Bulgarian).

3. Minieka, E., "Optimization Algorithms for Networks and Graphs, Marcel Dekker, Inc., New York and basel, 1978 /Майника, Э.Алгоритмы оптимизации на сетях и графах, М., "Мир" р1981/.

4. Christofides, N., graph Theory. An Algorithmic approach, Academic Press lnc /London/ Ltd. 1975, 1997 /Кристофидес, Н. Теория графов.Алгоритмический подход, М., "Мир", 1978/.

5. Swami, M., Thulasirman, Graphs, Networks and Algorithms, John Wiley & Sons, 1981 /Сваами М., К. Тхуласирман. Графы, сети и алгоритмы, М., "Мир", 1984/.

Abbreviation: **FS**: Fall Semester, **SS**: Spring Semester

INTRODUCTION IN LATEX 2e

Semester: 4

Course type: Lectures/Seminars

ECTS Credits: 4 credits

Course status: Selective Course in the Information systems end technology B.S. Curriculum

Course description: The Course is an Introduction in LATEX 2e used as a word processor when preparing text in mathematical, computer and other sciences.

Course aims: Non-trivial introduction in some important for Mathematicians and Computer scientists system for writing text with high quality.

Teaching methods: lectures, group discussions or workshop, projects, other methods

Requirements/ Prerequisites: Basic knowledge in Mathematics and Programming.

Materials: Textbook and manual of the course are published; access to web sites via Internet.

Evaluation: Written examination and discussion at the end of the semester, individual tasks and the general student's work during the semester.

Registration for the course: not necessary

Registration for the exam: in the department office, co-ordinated with the lecturer.

GRAPHIC DESIGN OF PRINTED AND PROMOTIONAL MATERIALS

Semester: 4

Course Type: seminars and lab exercises

Hours per week/SS: 1 seminar and 2 lab hours per week / SS

ECTS credits: 4 credits

Course Status: Optional Course.

Course Description: The course is a practical introduction to desktop publishing systems. Students learn the best practices in the development of print and electronic materials, such as brochures, leaflets, posters, magazines, newspapers and more. Studied are the principles of working with the software used in publishing. Discussed are typical problems in the field of publishing and advertising activities.

The course prepares students for the future development of different types of designs of promotional materials, web sites and more.

Course Objectives: This course aims to provide students with knowledge and additional training in the theory and practice of publishing systems. They will learn about the methods of digital image processing, how to create vector graphics and prepress of promotional materials with different purpose.

Teaching Methods: Lectures, demonstrations, work on project and teamwork.

Requirements: Needed basic knowledge of operating systems, information technology, graphics editors and working with multimedia files.

Assessment: Evaluating the student shall be carried out in the sixth grad scale – 2, 3, 4, 5, 6. Evaluation of current control is obtained by taking the average of the assessment of coursework and paper. Students who have a minimum average estimate /3/ of the current control is not allowed to test the regular session. They must present additional development and evaluation after receiving at least medium /3/ be admitted to the written examination of supplementary or liquidation session. The final estimate is derived from the average of the current control and evaluation of the written exam.

Registration for the Course: Submitted an application to the academic department at the end of current semester.

Registration for the Exam: Coordinated with the lecturer and Student Service Department.

References:

1. Rebecca Gagen, Kim Golombisky, White Space is Not Your Enemy: A Beginner's Guide to Communicating Visually through Graphic, Web and Multimedia Design, Focal Press, 2010
2. John McWade, Before & after graphics for Business, Peachpit Press, 2005
3. Roger C. Parker, Design to Sell: Use Microsoft® Publisher to Plan, Write and Design Great Marketing Pieces, Microsoft Press, 2006
4. Brian P. Lawler, Official Adobe Print Publishing Guide, Second Edition: The Essential Resource for Design, Production, and Prepress, Adobe Press, 2005
5. Elizabeth Eisner Reding (2013) Microsoft Publisher 2013: Illustrated, Cengage Learning Publishing
6. Joy L. Starks (2014) Microsoft Publisher 2013: Complete, Cengage Learning Publishing
7. Tamara Weinberg (2009) The new community rules. Marketing on the social web, O'Reilly Media
8. John DiMarco (2010) Digital Design for Print and Web. An Introduction to Theory, Principles, and Techniques, Wiley Publishing
9. Fazreil Amreen (2013) Instant GIMP Starter: Learn the basics of GIMP through practical examples, Packt Publishing
10. Web site of course,

http://194.141.86.222/lecture/rkraleva/LetenSem/IzdSystem/izdSys.html

11. SCRIBUS: Open Source Desktop Publishing, http://www.scribus.net/canvas/Scribus, 2012
12. GIMP: GNU Image Manipulation Program, http://www.gimp.org/, 2012
13. INSCAPE: Open Source Scalable Vector Graphics Editor, http://inkscape.org/, 2012

Abbreviation: SS: Spring Semester

PROGRAMMING WITH RUBY

Semester: 4

Course Type: seminars and lab exercises

Hours per week/SS: 1 seminar and 2 lab hours per week / SS

ECTS credits: 4 credits

Course Status: Optional Course.

Course Description: The course is a practical introduction to scripting object-oriented programming language Ruby with open-source. It combines parts of other programming language such as Perl, Smaltalk, Eiffle, Ada and Lisp. Ruby is the bases framework Ruby on Rails, which is one of modern languages to develop web applications. The course will examine various implementations of Ruby, as IronRuby, Ruby on Rails, RubyMine. Ruby is considered flexible because the language that programmers can freely modify any part of its code.

Course Objectives: This course aims to provide students with knowledge and additional training in the theory and practice of programming language Ruby. Will be discussed at the method of operating with numeric types, arrays, proc, functions, and classes and metaprogramming. During the seminars will be discussed many examples.

Teaching Methods: Lectures, demonstrations, work on project and teamwork.

Requirements: Needed basic knowledge of introduction to programming and web systems and technologies.

Assessment: Evaluating the student shall be carried out in the sixth grad scale – 2, 3, 4, 5, 6. Evaluation of current control is obtained by taking the average of the assessment of coursework, paper and the average grade of the tasks solved during the laboratory sessions. Students who have a minimum average estimate /3/ of the current control is not allowed to test the regular session. They must present additional development and evaluation after receiving at least medium /3/ be admitted to the written examination of supplementary or liquidation session. The final estimate is derived from the average of the current control and evaluation of the written exam.

Registration for the Course: Applied to the academic department at the end of current semester.

Registration for the Exam: Coordinated with the lecturer and Student Service Department.

References:

1. Huw Collingbourne (2011) The book of Ruby. A Hands-on guide for the adventurous, No starch Press, San Francisco
2. Paolo Perrotta (2010) Metaprogramming Ruby, O'Reilly
3. Gregory T. Brown (2009) Ruby Best Practices, O'Reilly
4. Adam Gamble, Cloves Carneiro Jr., Rida Al Barazi (2013) Beginning Rails 4, Third Edition, Apress
5. David Copeland (2013) Build awesome command-line applications in Ruby 2. Control your computer, simplify your life, The Pragmatic Bookshelf, USA

6. Sandi Metz (2013) Practical object-oriented design in Ruby, Addison-Wesley Press, USA
7. Ophir Frieder, Gideon Frieder, David Grossman (2013) *Computer science programming basics with Ruby*, O'Reilly Publishing
8. Carleton DiLeo (2019) *Clean Ruby: A Guide to Crafting Better Code for Rubyists*, Apress.
9. David A. Black, Leo Joseph (2019) *The Well-Grounded Rubyist*, 3rd Edition, MANNING Shelter Island Press.
10. Jay Gods (2018) *Ruby Data Processing: Using Map, Reduce, and Select*, Apress.
11. Bala Paranj (2017) *Test Driven Development in Ruby. A Practical Introduction to TDD Using Problem and Solution Domain Analysis*, Apress.
12. Malay Mandal (2017) *Ruby Recipes. A Problem-Solution Approach*, Apress.
13. Peter Cooper (2016) *Beginning Ruby*, 3rd Edition, Apress.
14. Stefan Wintermeyer (2018) *Learn Rails 5.2: Accelerated Web Development with Ruby on Rails*, Apress.
15. John S. Conery (2011) *Explorations in Computing. An Introduction to Computer Science*, CRC Press.
16. John S. Conery (2011) *Explorations in Computing. An Introduction to Computer Science*, CRC Press.
17. Paolo Perrotta (2014) *Metaprogramming Ruby 2. Program Like the Ruby Pros*, The Pragmatic Programmers, LLC.

Abbreviation: SS: Spring Semester

COMBINATORICS, CODING THEORY, CRYPTOGRAPHY

Semester: **4**

Form of the course: **Lectures/exercises**

Hours (per week): **3 hours lectures + 3 hours exercises per week, summer semester**

Credits: **11 credits**

Status of the course in the educational plan:

The course is to be chosen in the educational plan of Informatics BSc curriculum.

Description of the course:

The course starts with introduction of the main notions of the Coding theory – error-correcting codes, Hamming distance, code parameters, and equivalency of codes. Then the necessary algebraic background (finite fields and vector spaces over finite fields) is developed and encoding and decoding with linear codes (including syndrome decoding) are studied. Important classes of codes are introduced and the theory of cyclic codes is developed. In the cryptographic part the classical chiphers are considered and followed by the modern systems for secret and public keys.

Scope of the course:

Obtaining knowledge of the theoretical backgrounds and practical abilities for applications of the Coding theory and the cryptography. Development of abilities for work with (linear) codes over finite field with special emphasis of their algebraic and combinatorial properties.

Methods: lectures, discussions, practical exercises of the codes under consideration

Preliminary requirements: The students must have basic knowledge from the Number theory and algebra.

Evaluation: permanent control during the semester (two written exams) and exam in the semester's end in two parts – problems solving and answering theoretical questions.

Registration for the course: by application in the Educational Office in the end of the semester

Registration for exam: up to agreement with the teacher and the Educational Office

Literature:

3. Notices (www.moi.math.bas.bg/~peter)

4. Raymond Hill. A First Course in Coding Theory, Calderon press, Oxford, 1986.

MATHEMATICAL MODELS IN ECONOMICS

Semester: 4

Course Type: **lecture, exercises;**

Hours per week/FS/SS: 2 lecture; 1 exercise week/ SS

ECTS credits: 5.5

Course Status: Optional course in the Computer Science BSc curriculum

Course Description:

Mathematical models in economics are a new topic in mathematics. In this course we present some element of the optimization theory, discrete optimization and probability theory.

Course Aims: Students should obtain knowledge and skills for matroid theory.

Teaching Methods: lectures, demonstrations and work on project

Requirements/Prerequisites: Linear algebra.

Assessment: course project

Registration for the Course: by request at the end of the current semester

Registration for the Exam: coordinated with the lecturer and Student Service Department

References:

Basic Titles:

D. J. A. Welsh. Matroid Theory. Acad. Press. New York. 1976

Вейль. Г. Элементарная теория выпуклых многогранников. В кн. Матричые игры. М: Физматгиз, 1966.

Abbreviation: SS: Spring Semester, **FS**: Fall Semester

MATROID THEORY

Semester: 4

Course Type: lecture, exercises;

Hours per week/FS/SS: 2 lecture; 1 exercise week/ SS

ECTS credits: 5.5

Course Status: Optional course in the Computer Science BSc curriculum

Course Description: Basic questions in the theory of matroids are discussed. The course introduces students to basic concepts related to matroids and combinatorial extreme problems where they find application..

Course Aims: Students should obtain knowledge and skills for matroid theory.

Teaching Methods: lectures, and exercises.

Requirements/Prerequisites: Knowledge in Mathematics.

Exam: final exam.

Registration for the Course: by request at the end of the current semester

Registration for the Exam: coordinated with the lecturer and Student Service Department

References:

Basic Titles:

1. Ding-Zhu Du, Panos M. Pardalos, Zhao Zhang. Nonlinear Combinatorial Optimization, Springer International Publishing, 2019
2. Matroid Theory and its Applications, Springer-Verlag Berlin Heidelberg, 2011.

Additional

1. D. J. A. Welsh. Matroid Theory. Acad. Press. New York. 1976

Abbreviation: SS: Spring Semester

SEPARABLE SETS OF VARIABLES

Semester: 4

Course Type: lectures and seminar exercises

Hours per week: 2 lecture hours and 1 seminar hour per week/SS

ECTS credits: 5.5

Course status: Optional course of Informatics BSc Curriculum.

Course Description: The study of behavior of discrete functions when they are substituted by some of their variables by constants, as well as identification of variables, is connected with the development of mathematical cybernetics, automata theory, it is used in the study of

recursive functions, in some methods for synthesis of control systems, in the synthesis of contact schemes (the cascade method) and many other parts of the theoretical and applied informatics. In the proposed course, some topics connected with separable sets of variables for the functions are considered, whose applied interpretation in the synthesis of contact schemes and automata make them actual and important. Graphs of functions with respect to separable pairs of their variables are studied. Great attention is paid to strongly essential and *c*-strongly essential variables for the functions.

Course Aims: Students should obtain basic knowledge in the area of Theory of separable sets of variables for the functions, as well as the ability to apply this theoretical knowledge to Graph Theory in respect to their separable pairs of variables.

Teaching Methods: lectures, exercises, course works and extracurricular occupation.

Requirements/Prerequisites: Preliminary knowledge of Graph Theory and Discrete Functions is useful.

Assessment: Current assessment (four homeworks H_1, H_2, H_3, H_4; two control works C_1, C_2 /course works/) and Exam grade.

Final Grade: $FG=0.05(H_1+H_2+H_3+H_4)+0.2(C_1+C_2)+0.4(\text{Exam})$

Registration for the Course: by request at the end of the previous semester.

Registration for the Exam: coordinated with the lecturer and Student Service Department.

References:

1. Чимев К. Н., Отделими множества от аргументи на функциите. Благоевград, 1982, 1-206.

2. Чимев К. Н., Функции и графи. Благоевград, 1983г, 1-195.

3. Chimev K., Separable Sets of Arguments of.Function, MTA Sz TAKI, Budapest, 1986, 1-173.

4. Чимев К. Н., И. Д. Гюдженов, Подфункции и мощност на някои класи функции, Благоевград, 1987, 1-220.

FUNCTIONAL PROGRAMMING

Semester: **5**

Type of Course: **Lectures and tutorials in computer lab**

Hours per week - **2 hours lectures and 2 hour tutorials in computer lab**

Credits Numbers: **4,0 credits**

Course Status: Core course in curriculum of major Informatics, Bachelor degree.

Course description:

The course is introduction in design and programming in Scheme LISP dialect.

Objectives:

The student should obtain knowledge of:

- Design and programming in Scheme.
- Practical aspects of functional programming.

Methods of teaching: seminars, tutorials, discussions, project based method.

Pre- requirements: C++ programming and Data Structure

Assessment and Evaluation

Tutorial - 30%

Final Test- 70%

The course is successful completed with at least 51% of all scores.

Registration for the Course: not required (core course)

Registration for the Exam: coordinated with the lecturer and the Student Service Office

References

1. R. Kent Dybvig / The Scheme Programming Language, Fourth Edition Copyright © 2009 The MIT Press. Electronically reproduced by permission. Illustrations © 2009 Jean-Pierre Hébert ISBN 978-0-262-51298-5 / LOC QA76.73.S34D93
2. Harold Abelson, Gerald Jay Sussman and Julie Sussman, Structure and Interpretation of Computer Programs, The MIT Press, 2 ed. 1996
3. Eisenberg, M. Programming in Scheme, MIT Press, 1990
4. Springer, G., D. Friedman. Scheme and the Art of Programming, MIT Press, 1989
5. Kurt Nørmark, Functional Programming in Scheme With Web Programming Examples, Department of Computer Science, Aalborg University, Denmark URL http://people.cs.aau.dk/~normark/prog3-03/html/notes/theme-index.html
6. Hailperin M., Kaiser B. , Knight K., Concrete Abstractions, Brooks/Cole Publishing Company, 1999
7. Harley, B., Wright, M., Simply Scheme: Introducing Computer Science, The MIT Press, 2 ed. 1999, URL http://www.cs.berkeley.edu/~bh/ss-toc2.html#implement-hof

NUMERICAL ANALYSIS

Semester: 5 semester

Course Type: lectures and lab exercises

Hours per Week/FS/SS: 3 lecture hours and 2 lab hours per week/FS

ECTS Credits: 8 credits

Course Status: Compulsory Course in the Informatics B.S. Curriculum

Course Description: The course in Numerical Analysis includes basic numerical methods of mathematical analysis, algebra, and differential equations: interpolation and least squares data fitting as methods for approximating functions given by tabulated data; numerical differentiation; numerical integration – Newton-Cotes and Gauss quadrature formulas;

numerical solution of nonlinear equations; numerical solution of linear systems of algebraic equations; numerical solution of the initial-value problem for ordinary differential equations of first order; numerical solution of the boundary value problem for ordinary differential equations of second order; and variational methods for solving operator equations (including differential equations).

Course Objectives: Students should obtain knowledge and skills for numerical solutions of problems in the area of mathematical analysis, algebra and differential equations, which are applicable for solving various problems.

Teaching Methods: lectures and lab exercises

Requirements/Prerequisites: Mathematical Analysis, Linear Algebra, Analytic Geometry, Differential Equations

Assessment: written final exam covering problems /omitted in case the average grade of two current problem tests is higher than Very Good 4.50/ (grade weight is 30 %) and theory on two topics (grade weight is 30 %); two homework (grade weight is 20 %) and two projects (grade weight is 20 %)

Registration for the Course: not necessary

Registration for the Exam: coordinated with lecturer and Student Service Department

References:

Basic Titles:

1. Yordanka Angelova – „Numerical Analysis for BSc students", Chemical, Technological and Metallurgical University, Sofia, 2006 (in Bulgarian).
2. D. T. Boyadzhiev, Snezhana Gocheva-Ilieva, I. V. Makrelov, L. I. Popova – „Numerical Analysis Handbook", Part I, 3-rd ed., ExPress, Gabrovo, 2010 (in Bulgarian).
3. D. T. Boyadzhiev, Snezhana Gocheva-Ilieva, L. I. Popova – „Numerical Analysis Handbook", Part II, Demetra, Sofia, 2012 (in Bulgarian).
4. B. Boyanov – "Lectures on Numerical Analysis", Darba Publishing House, Sofia, 1995 (in Bulgarian).
5. Snezhana Gocheva-Ilieva – "Computer Numerical Analysis", Paisii Hilendarski Plovdiv University Press, Plovdiv, 2013 (in Bulgarian) (also available online).
6. Stefka Dimova, Tatiana Chernogorova, Angelina Yotova – "Numerical Methods for Differential Equations", St. Kliment Ohridski Sofia University Press, Sofia, 2010 (in Bulgarian).
7. Konstantin Kazakov – „Finite elements method for modeling building constructions", Prof. Marin Drinov Academic Press, Sofia, 2010 (in Bulgarian).
8. M. Kaschiev – "Numerical Analysis Handbook", Martilen Publishing House, Sofia, 1994 (in Bulgarian).
9. "Numerical Analysis Problem Book", 2-nd ed., St. Kliment Ohridski Sofia University Press, Sofia, 1994 (in Bulgarian).
10. V. Pasheva – "Introduction to Numerical Analysis", Technical University-Sofia, 2009.

11. Bl. Sendov, V. Popov – "Numerical Analysis", Part I, Kliment Ohridski Sofia University Press, Sofia, 1996; Part II, Nauka and Izkustvo Publishing House, Sofia, 1978 (in Bulgarian).

Additional Titles:

12. R. L. Burden, J. D. Faires – "Numerical Analysis", ", 9-th ed., Cengage Learning, Stamford, CT, USA, 2010.
13. Rizwan Butt – "Introduction to Numerical Analysis using Matlab", Jones and Bartlett Publishers, Sudbury, MA, USA, 2009.
14. C. D. Conte, Carl de Boor – "Numerical Analysis: An Algorithmic Approach", 3-rd ed., McGraw Hill Education, Columbus, OH, USA, 2005.
15. J. D. Faires, R. L. Burden – "Numerical Methods", 4-th ed., Brooks/Cole Publishing Company, Pacific Grove, CA, USA, 2013.
16. Timothy Sauer – "Numerical Analysis", 2-nd ed., Pearson Education, London, 2011.
17. S. M. Stefanov – "Numerical Analysis", MS4004-2203, Limerick, 1998.
18. William F. Trench – "Elementary Differential Equations with Boundary Value Problems. Student Manual", Trinity University, San Antonio, Texas, USA, 2013 (also available online).

Abbreviation: FS: Fall Semester SS: Spring Semester

SPECIALIZED STATISTICAL SOFTWARE

Semester: 5 semester

Type of Course: lectures, and tutorials in computer lab

Hours per week: 2 hours lectures, and 2 hours tutorials in computer lab/winter semester

Credits Numbers: 5.5 credits

Course Status: Optional course.

Course description:

The course Specialized software /Statistical analysis of data with the help of IT (MS Excel, Statistica, SPSS)/ should introduce students to apply the methods of statistics in practice with the tools of IT.

The main objectives of the modeling the empirical data and application in the IT are introduced in the course. Contemporary technologies used for their application (MS EXCEL, SPSS and STATISTICA) are also included in the course.

Objectives:

- To give students theoretical knowledge of the main statistical procedures, as well as some specifics of their usage.
- To teach students how to create models for statistical analysis of experimental data.

- To present contemporary IT for statistical analysis to the students.
- To prepare students for their future researches.

After successfully completing the course the students should:

- know and be able to apply procedures for statistical analysis of experimental data;
- manage to create, edit, export and import data in contemporary IT;
- be able to create models for statistical analysis of experimental data.

Methods of teaching: seminars, tutorials, discussions, project based method, simulations

Pre- requirements: Probability and Statistics, Information Technology

Assessment and Evaluation

Project- 30%

Final Test- 30%

Individual students works-40%

The course is successful completed with at least 50% of all scores.

Registration for the Course: required

Registration for the Exam: coordinated with the lecturer and the Student Service Office

THEORETICAL FOUNDATIONS OF INFORMATICS

Semester: 5
Course type: lectures
Hours (weekly) / FS / SS: lectures – 2 hours per week + seminars – 1 hour per week/ FS
Number of ECTS credits: 5.5
Type of the course in the curriculum:
Compulsory course from the curriculum of the "Informatics" Bachelor's degree programme

Course description: It deals with the theory of algorithms.

Goal: The course in Theoretical foundations of informatics aims to introduce students to the basic concepts and results of the theory of algorithms.

Teaching methods: lectures, demonstrations, problem solving.

Prerequisites: The acquired knowledge is useful in theory of algorithms.

Examination and assessment procedures:

The estimation of the acquired knowledge is based on a written exam which consists of problem solving and theoretical knowledge examination (writing on a topic from the syllabus provided to students)

The final grade includes the assessment of the students' progress throughout a course of study (30 %) plus the examination at the end of it (70 %)

Course enrolment: Students should submit an application at the academic affairs department at the end of the current semester

Registration for examination: coordinated with the lecturer and the academic affairs department

References:

1. Н. Катленд. „Вьчислимость, введение в теорию рекурсивных функций".

2. Барвайс. "Энциклопедия по математической логике".

Notes: FS: Fall semester

CRISES AND DISASTERS

Semester: 5

Kind on rates: practice exercises

Hours/the seventh/: 2 hours exercises

Number the credit: 2 credits

Course status: Compulsory discipline from educational the plan of speciality "Informatics".

Description of discipline:

Teaching on education discipline includes make a study of underlying oppearans a natural calamity, accident and mass crash. Ways and means management crisis from mass character, means for protection and accordance winner help on suffer people.

Aims of discipline:

The students have got to acquire knowledge for underlying oppearans a natural calamity, ways and means for management crisis from mass character, means for protection, computers and other electric truck system necessary for management.

Teaching methods: Practice demonstration.

Advance conditions: Desirable on anatomy, ecology and resources for electric truck management.

Appraisement: Written examination.

Registration for examination: Coordinated with the lecturer and student service department

Registration for the course: not necessary

Refferences:

1. Direkov, L. Protection of man and environment protection at extreme conditions.1996.

OPERATIONS RESEARCH

Semester: 5

Course Type: lectures, seminars

Hours per week/FS/SS: 2 lecture; 1 seminar exercise week/ FS

ECTS credits: 5

Course Status: Optional course in the Computer Science

Course Description:

Questions from the theory of operations research are considered. In the course, students will learn how to find the optimal solution to a given problem. The basic concepts of operations research are given and the approach is algorithmic.

Course Aims:

Students should obtain knowledge and skills to find the optimal solution in the analyzing problem**;**

Teaching Methods: lectures, demonstrations and work on project

Requirements/Prerequisites: Klowledge in Mathematics.

Exam: final exam

Registration for the Course: by request at the end of the current semester

Registration for the Exam: coordinated with the lecturer and Student Service Department

References:

Basic Titles:

1. Michael W. Carter, Camille C. Price, Ghaith Rabadi. Operations Research: A Practical Approach. CRC press, 2019
2. Венцель И. "Исследование операций". Москва, 1970.
3. Vagner G. "Operational Research", Vol. I-III 1998.

Additional:

1. Hamdy A. Taha – "Operations Research. An Introduction", 9-th ed., Prentice Hall, USA, 2010
2. S. M. Stefanov – "Separable programming. Theory and Methods:, Kluwer Academic Publishers, Dordrecht-Boston-London, 2001.

Abbreviation:

FS: Fall Semester

SS: Spring Semester

MATHEMATICAL FUNDAMENTALS OF COMPUTER GRAPHICS

Semester: 5

Course Type: lectures, seminars

Hours per week/FS/SS: 2 lecture, 1 labs week/FS

ECTS credits: 5

Course Status: Optional course in the Computer Science

Course Description: The course introduces a theoretical and practical introduction to the mathematical fundamentals of computer graphics. Basic concepts, terms, approaches, and methods used in computer graphics are presented. This knowledge would support the development of software applications related to computer graphics, and the modelling of real-world objects and processes.

The course is adapted towards the students of the "Informatics" specialty, and a good mastery of the proposed material is essential for the creation of high-quality computer games and computer models of real-world objects.

It is a continuation of the courses in Computer Design of Printed and Promotional Materials, Computer Mathematics, and Mathematical Analysis.

Course Objectives: This course aims to provide knowledge of the students about the mathematical methods and algorithms used in computer graphics and their implementation in real software applications.

Teaching Methods: lectures, demonstrations and work on project

Requirements/Prerequisites: Basic knowledge of programming, computer mathematics, and mathematical analysis are required.

Assessment: Evaluating the student shall be carried out in the sixth grad scale – 2, 3, 4, 5, 6. Evaluation of current control is obtained by taking the average of the assessment of coursework and paper. Students who have a minimum average estimate /3/ of the current control is not allowed to test the regular session. They must present additional development and evaluation after receiving at least medium /3/ be admitted to the written examination of supplementary or liquidation session. The final estimate is derived from the average of the current control and evaluation of the written exam.

Registration for the Course: by request at the end of the current semester

Registration for the Exam: coordinated with the lecturer and Student Service Department

References:

1. John Vince (2017) "Mathematics for Computer Graphics" (Undergraduate Topics in Computer Science), 5th Edition, Springer.
2. Стоян Малешков, Веселин Георгиев (2014) "Компютърна графика и фотореалистична визуализация", Нов български университет, София.
3. Eric Lengyel (2012) "Mathematics for 3D Game Programming and Computer Graphics", 4th Edition, Cengage Learning PTR.
4. Peter Shirley, Steve Marschner (2018) "Fundamentals of Computer Graphics", CRC Press.
5. Frank Klawonn (2008) "Introduction to Computer Graphics Using Java 2D and 3D", Springer.
6. Donald Hearn, M. Pauline Baker (1996) "Computer Graphics C Version", 2nd Edition, Prentice Hall

7. Mike Bailey, Steve Cunningham (2016) "Graphics Shaders: Theory and Practice", 2nd Edition, CRC Press.
8. John Collomosse (2008) "Fundamentals of Computer Graphics - CM20219", Lecture Notes, University of Bath, UK.
9. John F. Hughes, Andries Van Dam, Morgna McGuire, David F. Sklar, James D. Foley, Steven K. Feiner, Kurt Akeley (2014) "Computer Graphics: Principles and Practice", 3rd Edition, Addison Wesley.
10. Steven J. Janke (2015) "Mathematical structures for computer graphics", John Wiley & Sons, Inc.
11. Fabio Ganovelli, Massimiliano Corsini, Sumanta Pattanaik, Marco Di Benedetto (2015) "Introduction to Computer Graphics. A practical Leaning Approach", CRC Press.
12. Edward Angel, Dave Shreiner (2012) "Interactive Computer Graphics. A Top-Down Approach with Shader-Based OpenGL®", 6th Edition, Addison-Wesley

Abbreviation: **FS**: Fall Semester **SS**: Spring Semester

PROGRAMMING WITH OBJECT PASCAL AND DELPHI

Semester: 5

Type of Course: lectures and labs

Hours per week - 2 lectures + 1 labs per week/FS

Credits: 5

Course Status: Optional course from the Computer Science Bachelor Curriculum.

Course description:

In the course students are introduced wiht methods and means of Object-oriented programing in integrated development interface for visual programing - Delphi. The students should have a basic knowledge on programming with Pascal. Suppouse that students are successablity passed courses in Programming and Data structures and Object-oriented programming (in SWU this courses are basic on program language C++) and students are known for fundamental skils in programming. In the course students develop programs using different platform and language - Object Pascal and Dlephi.

Objectives:

Basic objectives and tasks:

- The students give knowledge for algorithum thinking;
- to give knowledge for Data stuctures, that can process with computer;

- to give knowledge for methods and skils in Object-oriented programming in integrated development environment for visual programming;
- to give knowledge for syntax of another program language (Object Pascal and Delphi);
- to give knowledge for good style in programming;
- to give knowledge for basic principles when develop applications.

Methods of teaching: lectures and labs

Pre- requirements: Basic knowledge in "Programming and Data structures".

Exam: two course projects and final exam

Registration for the Course: A request is made by students at the end of the current semester

Registration for the Exam: Coordinated with the lecturer and the Student Service Office

References

1. Франк Елер *Delphi 6.* „ИнфоДАР", 2001.

2. Христо Крушков *Програмиране с Delphi.* Пловдив, „Макрос", 2004.

3. Мартин Гардън *Delphi – създаване на компоненти.* АмПрес, 1999.

4. Хавиер Пачеко, Стийв Тейхера *Delphi 5,* т.1, т.2, т.3, „ИнфоДар", 1999

5. Хавиер Пачеко, Стийв Тейхера *Delphi 5 – ръководство за напреднали.* „ИнфоДАР", 1999.

6. Марко Канту, *Mastering Delphi 6*, т.1, т.2, "Софтпрес", 2002

Abbreviation:

FS: Fall Semester

SS: Spring Semester

MATHEMATICAL THEORY OF DATABASE

Semester: 5

Course Type: lecture

Hours per week/FS/SS: 2 lecture; 1 exercise week/FS

ECTS: 5 credits:

Department: Department of Computer Science, telephone: 073 588 532

Course Status: Optional course in the Computer Science

Course Description:

In this course we present the differences between the traditional, no database approach to information system design and the database approach.

Course Aims:

Students should obtain knowledge and skills for designing of real database**;**

Teaching Methods: lectures, demonstrations and work on project

Requirements/Prerequisites: Linear algebra, Computer languages.

Assessment: course project

Registration for the Course: by request at the end of the current semester

Registration for the Exam: coordinated with the lecturer and Student Service Department

References:

Basic Titles:

1. Pavel Azalov. Database. Relation and objective approach, Tehnika, 1991 г.
2. J .C. Shepherd, Database Management: Theory and application. 1990, Boston

Abbreviation:

FS: Fall Semester

SS: Spring Semester

PROGRAMMING WITH C++ BUILDER

Semester: 5

Type of Course: lectures and labs

Hours per week - 2 lectures + 1 labs per week/FS

Credits: 5

Course Status: Optional course from the Computer Science Bachelor Curriculum.

Course description:

In the course students are introduced with methods and means of Object-oriented programming in integrated development interface for visual programming – C++ Builder. The students should have a basic knowledge on programming. Suppose that students are success passed courses in Programming and Data structures and Object-oriented programming (in SWU this courses are basic on program language C++) and students are known for fundamental skills in programming. In the course students develop programs using different platform – C++ Builder.

Objectives:

Basic objectives and tasks:

- The students give knowledge for algorithm thinking;
- to give knowledge for Data structures, that can process with computer;
- to give knowledge for methods and skills in Object-oriented programming in integrated development environment for visual programming;

- to give knowledge for good style in programming;
- to give knowledge for basic principles when develop applications.

Methods of teaching: lectures and labs

Pre- requirements: Basic knowledge in „Programming and Data structures" and „Object-oriented programming".

Exam: two course projects and final exam

Registration for the Course: A request is made by students at the end of the current semester

Registration for the Exam: Coordinated with the lecturer and the Student Service Office

References

1. Richard Kaiser, A.v. Biljon, C.Y. Crocker, and P. Lietmeyer. C++ with Borland C++Builder: An Introduction to the ANSI/ISO Standard and Object-Oriented Windows Programming, 2008

2. Jarrod Hollingworth, Bob Swart, Mark Cashman, and Paul Gustavson. Borland C++ Builder 6 Developer's Guide, 2002

3. Satya Sai Kolachina. C++ Builder 6 Developers Guide with CDR (Wordware Delphi developer's library), 2002

4. Borland C++ Builder: The Complete Reference by Herbert Schildt and Gregory L. Guntle (Paperback - April 25, 2001)

5. John Miano, Thomas Cabanski, and Harold Howe. Borland C++ Builder: the Definitive C++ Builder Problem Solver, 1997

6. Kent Reisdorph and Ken Henderson. Sams Teach Yourself Borland C++ Builder in 21 Days, 1997.

Abbreviation:

FS: Fall Semester

JAVASCRIPT PROGRAMMING

Semester: 5

Type of Course: lectures and tutorials in computer lab.

Hours per week: 2 hours lectures and 1 hours tutorials in computer lab/ spring semester.

Credits Numbers: 5 credits

Course Status: Elective course.

Course description: The course is introduction in design and development of JavaScript applications and interfaces of Web based information systems.

Objectives:

The student should obtain knowledge of:

- Design and implementation of JavaScript applications.

- Design and implementation JavaScript client interfaces of Web-based information systems.

Methods of teaching: lectures, tutorials, discussions, project based method.

Pre- requirements: Database systems (core course).

Assessment and Evaluation

Project- 50%

Final Test- 50%

The course is successful completed with at least 51% of all scores.

Registration for the Course: by request at the end of the current semester

Registration for the Exam: coordinated with the lecturer and the Student Service Office

References:

1. Денис Колисниченко, JavaScript & jQuery - практическо програмиране, Асеневци, 2014
2. Stefanov, Stoyan, Object-Oriented JavaScript, Packt Publishing, 2008
3. Zakas, N., The Principles of Object-Oriented JavaScript, No Starch Press, 2014

DOMAIN SPECIFIC LANGUAGES

Semester: 5

Course Type: lectures and labs

Hours (weekly)/WS/SS: 2 lecture and 1 labs per week / WS

ECTS Credits: 5 credits

Course Status: Optional course.

Short Description: The course will introduce basic methods for creating language extensions - heterogeneous and homogeneous domain-specific languages. Some of the popular external DSL, and tools to create them will be addressed. By using the so-called. Framework processors in the course will be realized two domain-specific languages.

Course Aims: The course aims to students background with specialized training in the creation of domain-specific languages.

Teaching Methods: Lectures, Labs, Discussions, Project Based Methods

Requirements/Prerequisites: Knowledge in Programming Basics, Object Oriented Programming with C++/Java.

Exam: final exam

Course enrolment: Students should submit an application at the academic affairs department at the end of the current semester

Registration for the exam: Coordinated with lecturer and Students Service Department

References:

1. Markus Voelter, DSL Engineering. Designing, Implementing and Using Domain-Specific Languages, http://dslbook.org, 2013
2. Martin Fowler, Domain Specific Languages, Addison-Wesley Professional, 2010
3. Debasish Ghosh, DSL In Action, Manning Publishing, 2011
4. Блог на Martin Flower http://martinflower.com
5. Markus Voelter, Generic Tools, Specific Languages, 2014
6. Martin Fowler, Patterns of Enterprise Application Architecture, Addison-Wesley, 2003
7. Cay Horstmann, Object-Oriented Design and Patterns, Wiley, 2006

Ivo Damyanov, Mila Sukalinska, DSL in practice, IJCA, Volume 115 (2), 2015

LOGIC PROGRAMMING

Semester: 6

Type of Course: lectures and labs

Hours per week - 2 lecture hours and 2 labs hours per week

Credits Numbers: 5,0

Course Status: Compulsory course from the Computer Science Bachelor Curriculum.

Course description:

The course provides introduction to logic programming. The main techniques of the structural approach of programming and their application using Prolog programming language are introduced.

Objectives:

The aim of the course is to teach the students with the techniques in development of algorithms and programmes using Prolog programming language. The knowledge will be used in the general theoretical, and some special courses for example programming for artificial intelligence

Methods of teaching: lectures and labs in a computer classroom

Pre-requirements: Basic knowledge in "Programming and Data structures" and "Mathematical Logics".

Exam: Written examination and discussion at the end of the semester, individual programming task and the general student's work during the semester.

Registration for the Course: not necessary

Registration for the Exam: Coordinated with the lecturer and the Student Service Office

References:

1. Dale Miller, Programming with Higher-Order Logic 1st Edition, 2012, ISBN-13:978-0521879408, ISBN-10: 052187940X

2. М. Тодорова. Езици за функционално и логическо програмиране, втора част Логическо програмиране. София, Сиела, 2003.

3. И. Держански, И. Ненова "Пролог за лингвисти." Tempus S-JEP-07272-94, 1997.

4. W. F. Clocksin, C. S. Mellish "Programming in Prolog" Springer-Verlag, 1984.

I. Bratko "Prolog Programming for Artificial Intilligence. Addison-Wesley,1986.

5. G. Metakides, A. Nerode "Principles of Logic and Logic Programming" Elsever, 1996.

6. John Malpas "Prolog: A Relational Language and its Application.Prentis-Hall,1987. Thayse, P. Gribomont, G. Louis, D. Snyers, P. Wodon, P. Goshet, E. Gregoire, E. Sanchez, Ph. Delsarte "Approshe Logique de L'Intelligence Artificielle. Paris, Bordas, 1988.

7. J. Doores, A. R. Reiblein, S. Vadera "Prolog – programming for tomorrow" Sigma Press, 1987.

PROBABILITY AND STATISTICS

Semester: 6

Type of Course: **lectures and labs**

Hours per week – **2 hours lectures, 2 hours tutorials in computer lab/winter semester**

Credits Numbers: **8 credits**

Course Status: obligatory course in curriculum of major Informatics. Bachelor degree.

Course description:

In this course, questions of Probability and Mathematical Statistics are considered. Algorithms connected with finding structural and numerical characteristics of graphs are represented. Basic notion of Probability and Statistics are considered connected with Theory of Estimations, and Decision Theory in case of big and small samples, testing of hypothesis based on models about the probability distributions of the features in the investigated population.

Objectives:

The students should obtain knowledge and understanding that the intercourse character needs to discover the connection Mathematics- Informatics- Physics- Economics and much more other sciences.

Methods of teaching: seminars, tutorials, discussions, project based method.

Pre-requirements: It is helpful the students have some knowledge in Analysis and Information Technology

Assessment and Evaluation

Three semestrials tests witch estimations will have part in the final estimation (50%)

The course is successful completed with at least 65% of all scores.

Registration for the Course: obligatory course

Registration for the Exam: coordinated with the lecturer and the Student Service Office

DATABASES

Semester: 6

Course Type: lectures and lab exercises

Hours per week/FS/SS: 3 lecture hours; 2 lab exercise hours per week/SS

ECTS credits: 8

Course Status: Compulsory course in the Computer Science

Course Description:

This course relies on primary readings from the database community to introduce graduate students to the foundations of database systems, focusing on basics such as the relational algebra and data model, schema normalization, query optimization, and transactions.

Course Aims:

Students should obtain knowledge and skills for designing of real database**;**

Teaching Methods: lectures, demonstrations and work on project

Requirements/Prerequisites: Knowledge in Mathematics.

Exam: final exam

Registration for the Course: no

Registration for the Exam: coordinated with the lecturer and Student Service Department

References:

1) Basic

1. Записки от лекции.
2. Павел Азълов. Бази от данни. Релационен и обектен подход, техника, 1991 г.
3. Юлиана Пенева, Бази от данни. I част. София, ИК "Регалия " 6, 2003 г.
4. Ullman, J., Widom, J., DATABASE SYSTEMS The Complete Book (2rd ed), Upper Saddle River, 2009, New Jersey.
5. Toby J. Teorey , Sam S. Lightstone , Tom Nadeau, H.V. Jagadish, Database Modeling and Design Database Modeling and Design, 2012, Morgan Kaufmann Press.
6. Rex Hogan. (2018) A Practical Guide to Database Design, CRC Press, USA.

2) Additional

7. Shepherd J.C. Database management: Theory and Application. Irwin Inc.,USA 1990.
8. Мейер Д.р Теория релационных баз данных. Издательство "Мир". 1987.

9. Vidya Vrat Agarwal, Beginning C Sharp 5.0 Databases, 2012 New York Press.

10. Alapati and Bill Padfield, Expert Indexing in Oracle Database, 2011, New York Press.

11. Henry H. Liu, Oracle Database Performance and Scalability A Quantitative Approach, 2011 A Jon Wiley and Son, US.

Abbreviation:

FS: Fall Semester

SS: Spring Semester

ALGORITHMS FOR GRAPHS AND NETWORKS

Semester: 6

Cours Tipe: Lectures and tutorials

Hours per week/FS/SS: 3 lecture hours, 1 lab exercise hour per week/SS

ECTS credits: 7 credits

Course Status: Obligatory course in the Computer Science B.S. Curricu

Course description:

In this course are considered some elements of the following main topics: introduction in graph theory (essential concepts and definitions. modeling with graphs and networks, data structures for networks and graphs; computational complexity; heuristics; tree algorithms (spanning tree algorithms. variations of the minimum spanning tree problem. branchings and arborescences); shortest-path algorithms (types of shortest-path problems and algorithms, shortest- paths from a single source, all shortest-path algorithms, the k- shortest-path algorithm, other shortest paths).

Goal:
Students should obtain basic knowledge and skills for solving optimization problems for graphs and networks.

Teaching methods: lectures, demonstrations, problem solving
Prerequisites: Mathematics, Informatics
Examination and assessment procedures: The estimation of the acquired knowledge is based on a written exam which consists of problem solving and theoretical knowledge examination (writing on a topic from the syllabus provided to students)
Course enrolment: Students should submit an application at the academic affairs department at the end of the current semester
Registration for examination: coordinated with the lecturer and the academic affairs department
References:

1. Ив.Мирчев, "Графи. Оптимизационни алгоритми в мрежи", Благоевград, 2001 г.
2. Ив.Мирчев, "Математическо оптимиране", Благоевград, 2000 г.
3. Evans J., Minieka, E., Optimization Algorithms for Networks and Graphs, Second Edition,, Inc., New York and Basel, 1992.

4. Erciyes K. Guide to Graph Algorithms: Sequential, Parallel and Distributed, Springer, 2018.
5. Goldengorin B. Optimization Problems in Graph Theory, In Honor of Gregory Z. Gutin's 60th Birthday Springer International Publishing AG, 2018.
6. Keijo Ruohonen. GRAPH THEORY. math.tut.fi/~ruohonen/GT_English.pdf
7. Ronald Gould. Graph Theory (Dover Books on Mathematics. 2012. US Cafifornia.
8. Lih-Hsing Hsu , Cheng-Kuan Lin, Graph Theory and Interconnection Networks. 1420044818;
9. Team DDU.Christofides, N., Graph Theory. An Algorithmic approach, Academic Press lnc (London) Ltd. 1975,
10. Swamy, M., K. Thulasirman, Graphs, Networks and Algorithms, John Wiley & Sons, 1981 (Сваами М., К. Тхуласирман. Графм, сети и алгоритми, М., "Мир", 1984).

Abbreviation: **SS**: Spring Semester

PRACTICAL COURSE IN DATABASES

Semester: 6

Course Type: lab exercises

Hours per week/SS: 2 labs hours per week/SS

ECTS credits: 2.0 credits

Course Status: Optional Course in Bachelor of Science Curriculum of Informatics

Course Description:

The course is practical and consolidate existing knowledge of students on the design of circuits of relational databases. Consider the process of normalization of relational schemes, as well as the physical creation of relational databases. Confirms is the syntax and use of basic SQL structures - create, insert, update, delete, select and others that are related to creating tables and relations between them, enter, update, delete and retrieval of data from relational databases. Particular attention is paid to the junction of the tables and work with views, as well as grouping of data and the use of aggregate functions. Finally considered milestones in the development of applications for working with databases, as well as designing and developing user interface applications of this type.

Course Objectives:

The aim of the course is to expand knowledge of databases, focusing on the use of applications and the creation of software for working with databases that have limited functionality. After completing the course students should be able to: - design schemes of relational databases; - Have skills to work with sql queries; - Have mastered various techniques and approaches for development of software in the field of databases.

Teaching Methods: Lectures, demonstrations, work on project and teamwork.

Requirements/Prerequisites: Needed basic knowledge of information technology, databases, object-oriented programming and work with MS Access. Desirable Knowledge of programming languages C + +, ObjectPascal and / or C #.

Assessment:

a) Current control

- Tasks for implementation during lab activities - n numbers by n <= number of exercises, ie Z1, Z2, ..., Zn;

- Extramural activity - production of paper - R to a given topic;

- Course Project - Development of design project K over a predetermined theme;

The current assessment (CC) = 0,5 * ((Z1 + Z2 + ... + Zn) / N) + 0,2 * R + 0,3 * K

Students with a current control > = 3.00 are examined;

b) Examination Assessment (EA) - written exam on a six-level system with an accuracy of 0.25;

c) Final assessment (FA) = 0.5 * (CC) + 0.5 * (EA);

The final result is a six-point scale: 2, 3, 4, 5 or 6.

Registration for the Course: Submitted an application to the academic department at the end of current semester.

Registration for the Exam: Coordinated with the lecturer and Student Service Department

References:

1. Andy Oppel. Databases: A Beginner's Gudie. McGraw-Hill. 2009

2. Rod Stephens. Beginning Database Design Solutions (Wrox Programmer to Programmer). Paperback, 2008

3. Ramez Elmasri and Shamkant Navathe. Fundamentals of Database Systems (6th Edition), Addison-Wesley; 6 edition (April 9, 2010)

4. Basit A. Masood-Al-Farooq. SQL Server 2014 Development Essentials. Packt Publishing. 2014

5. Gavin Powell. Beginning Database Design. Wiley Publishing, Inc. 2006.

6. Paul Wilton, John W. Colby. Beginning SQL. Wiley Publishing, Inc. 2005.

7. Thomas M. Connolly, Carolyn E. Begg. Database Systems - A Practical Approach to Design, Implementation, and Management (Fourth Edition). Pearson Education Limited 2005.

Abbreviation: SS: Spring Semester

PRACTICAL COURSE IN LOGIC PROGRAMMING

Semester: 6

Type of Course: lectures and labs

Hours per week - 2 labs hours per week

Credits Numbers: 2,0

Course Status: Optional course from the Computer Science Bachelor Curriculum.

Course description:

The course provides introduction to logic programming. The main techniques of the structural approach of programming and their application using Prolog programming language are introduced.

Objectives:

The aim of the course is to teach the students with the techniques in development of algorithms and programmes using Prolog programming language. The knowledge will be used in the general theoretical, and some special courses for example programming for artificial intelligence

Methods of teaching: lectures and labs in a computer classroom

Pre-requirements: Basic knowledge in "Programming and Data structures" and "Mathematical Logics".

Exam: Written examination and discussion at the end of the semester, individual programming task and the general student's work during the semester.

Registration for the Course: Submitted an application to the academic department at the end of current semester.

Registration for the Exam: Coordinated with the lecturer and the Student Service Office

References:

1. М. Тодорова Езици за функционално и логическо програмиране, втора част Логическо програмиране. София, Сиела, 2003.

2. И. Держански, И. Ненова "Пролог за лингвисти." Tempus S-JEP-07272-94, 1997.

3. W. F. Clocksin, C. S. Mellish "Programming in Prolog" Springer-Verlag, 1984.

4. I. Bratko "Prolog Programming for Artificial Intilligence. Addison-Wesley,1986.

5. G. Metakides, A. Nerode "Principles of Logic and Logic Programming" Elsever, 1996.

6. John Malpas "Prolog: A Relational Language and its Application.Prentis-Hall,1987.

7. A. Thayse, P. Gribomont, G. Louis, D. Snyers, P. Wodon, P. Goshet, E. Gregoire, E. Sanchez, Ph. Delsarte "Approshe Logique de L'Intelligence Artificielle. Paris, Bordas, 1988.

8. J. Doores, A. R. Reiblein, S. Vadera "Prolog – programming for tomorrow" Sigma Press, 1987.

WORKSHOP ON ASYNCHRONOUS AND PARALLEL PROGRAMMING WITH THE .NET FRAMEWORK

Semester: 6

Course Type: practical labs

Hours (weekly)/WS/SS: 2 labs per week / SS

ECTS Credits: 2 credits

Course Status: Optional course.

Short Description: Modern computer architectures and rich user interface impose new requirements on the performance and responsive interface. All of this nowadays is achieved through parallel architectures, multithreaded and asynchronous tasks execution. The course emphasizes the practical implementation with .NET Framework such as Parallel LINQ, TPL and others.

Course Aims: The course aim is to give theoretical and practical background to students on the principles of parallel and asynchronous processing with Microsoft .NET Framework.

Teaching Methods: Labs, Discussions, Project Based Methods

Requirements/Prerequisites: Knowledge in Programming Basics, Object Oriented Programming with C++/Java, Operating Systems.

Exam: final exam

Course enrolment: Students should submit an application at the academic affairs department at the end of the current semester

Registration for the exam: Coordinated with lecturer and Students Service Department

References:

1. Пламенка Боровска, Милена Лазарова, Паралелна информационна обработка: Системни архитектури, паралелни алгоритми, паралелно програмиране, Сиела, 2007
2. Alex Davies, Async in C# 5.0, O'Reilly Media, 2012
3. Rodney Ringler, C# Multithreaded and Parallel Programming, Packt Publishing, 2014
4. John Cheng, Max Grossman, Ty McKercher, Professional CUDA C Programming, Wrox Publishing, 2014
5. Светлин Наков и Веселин Колев, Въведение в програмирането със C#, Фабер, 2011, ISBN: 978-954-400-527-6
6. Eric Gunnerson and Nick Wienholt, A programmer's Guide to C# 5.0, APress, 2012
7. Daniel Solis, Illustrated C# 2012, 2nd.Edition. APress, 2012

Thuan Thai and Hoang Lam, .NET Framework Essentials, 2nd Edition, O`Reilly, 2002, ISBN 0-596-00302-1

PRACTICAL COURSE IN WEB DESIGN

Semester: 6

Course Type: lab exercises

Hours per week/SS: 2 labs hours per week/SS

ECTS credits: 2.0 credits

Course Status: Optional Course in Bachelor of Science Curriculum of Informatics

Course Description: The course examines issues and techniques related to the content organization and visualization on the web. Techniques for the static and dynamic pages developing and integrating them into complete websites are presented. An introduction to HTML, XHTML, and CSS is also provided. During the laboratory sessions, a website will be developed using the languages and technologies as HTML, CSS, JavaScript, C # and ASP.Net MVC. This course will allow the students to develop and refine the skills to design website designs and concepts. They can study how to use the appropriate fonts on the web and how to create and process vector and raster images suitable for web content.

Course Objectives: This course aims to provide depth theoretical knowledge and practical abilities in the field of adaptive web design. They will study the developing methods of websites, layout, and composition of the web elements, depending on the type of device, how to publish websites and support a web server.

Teaching Methods: Lectures, demonstrations, work on project and teamwork.

Requirements: Needed basic knowledge of operating systems, information technology, graphics editors and working with multimedia files.

Assessment: Evaluating the student shall be carried out in the sixth grad scale – 2, 3, 4, 5, 6. Evaluation of current control is obtained by taking the average of the assessment of coursework and tasks solved during the semester. Students who have a minimum average estimate /3/ of the current control is not allowed to test the regular session. They must present additional development and evaluation after receiving at least medium /3/ be admitted to the written examination of supplementary or liquidation session. The final estimate is derived from the average of the current control and evaluation of the written exam.

Registration for the Course: Submitted an application to the academic department at the end of current semester.

Registration for the Exam: Coordinated with the lecturer and Student Service Department.

References:

1. Giovanni Difeterici, *The Web Designer's Roadmap*, SitePoint, 2012
2. Jason Beaird, *The Principles of Beautiful Web Design*, SitePoint, 2010
3. Steve Fulton and Jeff Fulton, *HTML5 Canvas*, 2nd Edition, O'Reilly Media, 2013
4. Bill Scott and Theresa Neil, *Designing Web Interfaces*, O'Reilly Media, 2009

5. Lara Callender Hogan, *Designing for Performance*, O'Reilly Media, 2015
6. António Pratas, *Creating Flat Design Websites: Design and develop your own flat design websites in HTML*, Packt Publishing, 2014
7. Jörg Krause, *Introducing Web Developmen*t, Apress, 2016
8. Joshua Johanan, Talha Khan and Ricardo Zea, *Web Developer's Reference Guide*, Packt Publishing, 2016
9. Jason Gonzales, *Mobile First Design with HTML5 and CSS3*, Packt Publishing, 2013
10. Brian P. Hogan, *Web Design for Developers: A Programmer's Guide to Design Tools and Techniques*, The Pragmatic Bookshelf, 2009
11. Peter Gasston, *Multi-Device Web Development with HTML5, CSS3, and JavaScript*, No Starch Press, 2013
12. Clarissa Peterson, *Learning Responsive Web Design*, O'Reilly Media, 2014
13. Bill Evjen, Scott Hanselman, Devin Rader, *Professional ASP.NET 4 in C# and VB*, Wiley Publishing, 2010
14. Dafydd Stuttard and Marcus Pinto, *The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws*, Second Edition, Wiley Publishing, 2011
15. Alexis Goldstein, Louis Lazaris, and Estelle Weyl, *HTML5 & CSS3 for the Real World*, Sitepoint, 2015
16. Aditya Ravi Shankar, *Pro HTML5 Games: Learn to Build your Own Games using HTML5 and JavaScript*, 2nd Edition, Apress, 2017
17. A. Flanagan and S.M. Maniatis, *Intellectual Property on the Internet*, University of London, 2008; http://www.londoninternational.ac.uk/sites/default/files/intellectual_property_internet.pdf
18. WIPO, *The Enforcement of Intellectual Property Rights: A Case Book*, 2012; http://www.wipo.int/edocs/pubdocs/en/intproperty/791/wipo_pub_791.pdf
19. Денис Колисниченко, *HTML 5 & CSS 3 практическо програмиране за начинаещи*, изд. Асеневци, 2012
20. Жюстин Томас, *Програмиране на WEB дизайн*, изд. Нови знания, 2013
21. Алдениз Рашидов, *HTML, XHTML & CSS*, изд. Асеневци, 2012
22. Сергей Соколов, *CSS3 в примери*, изд. Асеневци, 2012

Abbreviation:

SS: Spring Semester

BITWISE OPERATIONS, GRAPHS AND COMBINATORIAL APPLICATIONS

Semester: 6

Type of Course: lectures

Hours per week – 1 lectures + 1 seminars

Credits Numbers: 2

Course Status: Elective course.

Course description: This elective course is the result of long work of the author as a scientist and lecturer in discrete mathematics and programming. It collected a number of scientific, scientific-methodological and applied research of the author in this field have been reflected in over 80 publications in leading journals. The idea of the discussed issues arise as we try to select appropriate examples in our work with talented students, yet touching to the modern scientific achievements. So naturally solving the current problem of the relationship between university education and modern science, we received the results, which definitely have a private scientific and practical value.

Aims and scopes: The aim of the course is to examine some combinatorial algorithms and their applications over binary matrices, finite sets and graphs.

Tasks:

- To provide interesting and meaningful examples where the use of bitwise operations increases the efficiency of algorithms;
- To make a mathematical model describing the structural diversity of different types of braiding the threads in the weaving of fabrics (weaving weaves). To solve some combinatorial problems related to these issues;
- To implement algorithms for operations on sets using bitwise operations;
- To realize the algorithm using operations on sets for solving any Sudoku;
- To study some combinatorial problems related to the puzzle Sudoku;
- With the help of set theory and relational algebra to make a mathematical model of any psychological testing;
- Two binary matrices are equivalent if one can be obtained from the other by moving some of the rows and / or columns. To investigate the factor-set on the so defined relation;
- To find a number of mutually disjoint pairs of S-permutation matrix;
- Describe some new graphs and set models in computer linguistics. Based on these models to describe a polynomial algorithm that verifies whether any context-free language is a subset of a group language.

Expected results:

- Proof of the fact that to be a good programmer required advanced mathematical knowledge and skills.
- Students to learn the algorithmic thinking;

- to form logical well individual steps in solving any task;
- to use Boolean algebra, matrix arithmetic graph theory and combinatorics in their future work;
- to assimilate data structures that can be processed by computer;
- to learn some already have become a classic algorithms, as well as to create their own algorithms;
- to evaluate the effectiveness of the algorithms, applying strictly mathematical proofs.

Methods of teaching: Lectures illustrated with school boards, slides, presentations, multimedia projector. Talks and discussions.

Pre-requirements: Students must have basic mathematical knowledge and skills in programming.

Exam: Written examination and discussion at the end of the semester, individual tasks and the general student's work during the semester.

Registration for the Course: it is necessary (an elective course)

Registration for the Exam: Coordinated with the lecturer and the Student Service Office

References:

A) Main:

1. К. Йорджев, Побитови операции, графи и комбинаторни приложения. ЮЗУ „Н. Рилски", Благоевград, 2014.

B) Additional:

1. K. Yordzhev (Iordzhev), An n^2 Algorithm for Recognition of Con\-text-free Languages. Cybernetics and Systems Analysis, 29, No.6 (1993) 922-927. http://link.springer.com/article/10.1007\%2FBF01122746
2. K. Yordzhev, An Entertaining Example of Using the Concepts of Context-Free Grammar and Pushdown Automation. Open Journal of Discrete Mathematics, 2 (2012), 105-108, http://www.scirp.org/journal/PaperInformation.aspx?PaperID=21127
3. K. Yordzhev, Some Combinatorial Problems on Binary Matrices in Programming Courses. Informational Technologies in Education, № 12, (2012) 39-43. http://ite.kspu.edu/en/issue-12/p-39-43
4. K. Yordzhev, Random Permutations, Random Sudoku Matrices and Randomized Algorithms. International Journal of Mathematical Sciences and Engineering Applications (IJMSEA), ISSN 0973-9424, Vol. 6, No. VI (2012), pp. 291-302. http://www.ascent-journals.com/ijmsea_contents_Vol6No6.html
5. K. Yordzhev, Bipartite Graphs Related to Mutually Disjoint S-permutation Matrices. ISRN Discrete Mathematics, vol. 2012, Article ID 384068, 18 pages, 2012. http://www.hindawi.com/journals/isrn.discrete.mathematics/2012/384068/
6. K. Yordzhev, A Representation of Context-free Grammars with the Help of Finite Digraphs. American Journal of Applied Mathematics. Vol. 1, No. 1 (2013) pp. 8-

11.
http://article.sciencepublishinggroup.com/pdf/10.11648.j.ajam.20130101.12.pdf

7. K. Yordzhev, A Representation of Context-free Grammars with the Help of Finite Digraphs. American Journal of Applied Mathematics. Vol. 1, No. 1 (2013) pp. 8–11.
http://article.sciencepublishinggroup.com/pdf/10.11648.j.ajam.20130101.12.pdf

8. K. Yordzhev, Inclusion of Regular and Linear Languages in Group Languages. International Journal of Mathematical Sciences and Engineering Applications, ISSN 0973-9424, Vol. 7 No. I (2013), pp. 323–336.

http://www.ascent-journals.com/ijmsea_contents_Vol7No1.html

9. K. Yordzhev, Bitwise Operations Related to a Combinatorial Problem on Binary Matrices. I. J. Modern Education and Computer Science, 4 (2013) 19-24. http://www.mecs-press.org/ijmecs/ijmecs-v5-n4/v5n4-3.html

10. K. Yordzhev, On the Number of Disjoint Pairs of S-permutation Matrices. Discrete Applied Mathematics, 161 (2013), 3072–3079.

http://www.sciencedirect.com/science/article/pii/S0166218X13002850

11. K. Yordzhev, The Bitwise Operations Related to a Fast Sorting Algorithm. International Journal of Advanced Computer Science and Applications, Vol. 4, No. 9 (2013) 103-107. http://thesai.org/Downloads/Volume4No9/Paper_17-The_Bitwise_Operations_Related_to_a_Fast_Sorting.pdf

12. K. Yordzhev, On an Algorithm for Obtaining All Binary Matrices of Special Class Related to V. E. Tarakanov's Formula. Journal of Mathematical Sciences and Applications, 1, no. 2 (2013): 36-38. http://pubs.sciepub.com/jmsa/1/2/5/jmsa-1-2-5.pdf

13. K. Yordzhev, On an Algorithm for Isomorphism Free Generations of Combinatorial Objects. International Journal of Emerging Trends & Technology in Computer Science, Vol. 2, No. 6 (2013) 215–220.

http://www.ijettcs.org/Volume2Issue6/IJETTCS-2013-12-21-080.pdf

14. K. Yordzhev, Fibonacci sequence related to a combinatorial problem on binary matrices. American Journal Mathematics and Sciences (AJMS), ISSN 2250 3102, Vol. 3, No. 1 (2014), 79–83. http://ajms.yolasite.com/resources/12.Fibonacci.pdf (preprint arXiv:1305.6790 http://arxiv.org/pdf/1305.6790v2.pdf)

15. K. Yordzhev, Factor-set of binary matrices and Fibonacci numbers. Applied Mathematics and Computation, 236, (2014), 235–238.

http://www.sciencedirect.com/science/article/pii/S0096300314004354

16. K. Yordzhev, On the probability of two randomly generated *S*-permutation matrices to be disjoint. Statistics & Probability Letters, 91 (2014).

http://www.sciencedirect.com/science/article/pii/S0167715214001370

17. I. Peneva, K. Yordzhev, Adnan Sharaf Ali, The Adaptation of Translation Psychological Test as a Necessary Condition for Ensuring the Reliability of Scientific Research. International Journal of Engineering Science and Innovative

Technology, Volume 2, Issue 4, July 2013, 557-560.

http://www.ijesit.com/Volume%202/Issue%204/IJESIT201304_71.pdf

18. K. Yordzhev, H. Kostadinova, On Some Entertaining Applications of the Concept of Set in Computer Science Course, Informational Technologies in Education. № 10 (2011), 24–29. http://ite.ksu.ks.ua/en/issue-10/p-24-29

19. K. Yordzhev, H. Kostadinova, Mathematical Modeling in the Textile Industry. Bulletin of Mathematical Sciences & Applications, Vol. 1, No. 1 (2012), 20–35.

http://www.bmsa.us/admin/uploads/00Czn3.pdf

20. K. Yordzhev, I. Peneva, Computer Administering of the Psychological Investigations – Set-Relational Representation, Open Journal of Applied Sciences, Vol 2, No 2, (2012) 110–114. http://www.scirp.org/journal/PaperInformation.aspx?PaperID=20331

21. K. Yordzhev, On an equivalence relation in the set of the permutation matrices. in Discrete Mathematics and Applications, SWU "N. Rilski", Blagoevgrad, Bulgaria, 2004, 77–87.

22. K. Yordzhev, On a Class of Binary Matrices. Mathematics and Educations in Mathematics, v.37 (2008), 245–250.

http://www.math.bas.bg/smb/2008_PK/2008/pdf/245-250.pdf

23. K. Yordzhev, An Example for the Use of Bitwise Operations in programming. Mathematics and education in mathematics, 38 (2009), 196–202.

http://www.math.bas.bg/smb/2009_PK/tom_2009/pdf/196-202.pdf

24. K. Yordzhev, On some numerical characteristics of a bipartite graph. Mathematics and education in mathematics, 43 (2014), 101–104.

http://www.math.bas.bg/smb/2014_PK/tom_2014/pdf/150-153.pdf

25. H. Kostadinova, K. Yordzhev, A Representation of Binary Matrices. in Mathematics and education in mathematics, 39, (2010), 198–206.

http://www.math.bas.bg/smb/2010_PK/tom/pdf/198-206.pdf

26. H. Kostadinova, K. Yordzhev, An Entertaining Example for the Usage of Bitwise Operations in Programming. Mathematics and natural science, v. 1, SWU "N. Rilski", (2011), 159-168. http://www.fmns.swu.bg/FMNS2011_Volume_1.pdf

27. D. Kovachev, K. Yordzhev, On Finding a Particular Class of Combinatorial Identities. Mathematics and Natural Science, v.1, 2009, 50-54.

http://www.fmns.swu.bg/Volume_1.pdf

28. I. Peneva, K. Gaidarov, K. Yordzhev, Computer Administering of Psychological Tests. Mathematics and Natural Sciences, v.1, SWU N. Rilsky, Blagoevgrad, 2009, 129-135. http://www.fmns.swu.bg/Fmns2009.html

29. G. Praskova, I. Petrov, K. Yordzhev, I. Peneva, Online Generation of Psychological tests. Mathematics and natural science-2013, Volume 1, SWU N. Rilsky, Blagoevgrad, Bulgaria, 2013, 235-240. http://www.fmns.swu.bg/FMNS2013-Volume_1.pdf

30. K. Yordzhev, H. Kostadinova, Mathematical Modeling of the Weaving Structure Design. Mathematics and education in mathematics, 39 (2010).

http://www.math.bas.bg/smb/2010_PK/tom/pdf/212-220.pdf

31. K. Yordzhev, I. Peneva, B. Kirilieva-Shivarova, A relational Model of Personality Psychological Tests. Mathematics and Natural Sciences, v.1, 2009, 69-77. http://www.fmns.swu.bg/Fmns2009.html

32. К. Йорджев, Х. Костадинова, Приложение на математически методи в сплиткознанието за получаване на количествени оценки на многообразието на тъкачните сплитки. Текстил и облекло, 1, (2011), 7–10.

33. И. Калчев, К. Йорджев, В. Вачков, Стохастични измервателни системи, Технически Университет, София, 2007.

34. И. Пенева, К. Йорджев, Интернет тестирането най-новата тенденция в психодиагностичните изследвания. Trakia journal of sciences, Volume 7, 2009, 155-159. http://tk.uni-sz.bg/files/M10_et_al.pdf

С. Щраков, К. Йорджев, М. Тодорова, Ръководство за решаване на задачи по дискретна математика. Благоевград, ЮЗУ "Н. Рилски", 2004.

MANAGEMENT AND FINANCING OF EDUCATIONAL AND SCIENTIFIC PROGRAMS

Semester: 6

Course Type: Lectures

Hours (weekly)/WS/SS: 2-hours' lectures / SS

ECTS Credits: 2 credits

Course Status: Optional course.

DESIGN AND DEVELOPMENT OF HUMAN COMPUTER INTERACTIONS

Semester: 6

Type of Course: Lectures and tutorials in computer lab.

Hours per week: 1 hour lectures and 1 hour tutorials in computer lab.

Credits Numbers: 2 credits

Course Status: elective course.

Course description: The course is directed to mastering of core princips and techniques for design, development and analysis of HCI. The problems as rules for graphical design of interface of software applications, psychological characteristics of users target groups, phycology of the colors etc., are discussed. Usability and accessibility of software applications are considered. The techniques for usability analysis are performed.

Objectives: The student should obtain basic knowledge in area of design, development and analysis of HCI.

Methods of teaching: lectures, tutorials, discussions, project based method.

Pre - requirements: No

Assessment and Evaluation

Project- 50%

Final Test- 50%

The course is successful completed with at least 51 % of all scores.

Registration for the Course: The students apply in Department of Informatics

Registration for the Exam: coordinated with the lecturer and the Student Service Office

References:

Basic

1. Стафанов, К. Проектиране на човеко-машинен интерфейс, онлайн курс - http://www-it.fmi.uni-sofia.bg/courses/HCI/index2.htm
2. Тупарова Д., Ползваемост на дигитални образователни ресурси, Образование и познание, София 2019

Additional

3. Shneiderman, B., & Plaisant, C. (2010). Designing the user interface: Strategies for effective human-computer interaction. Boston: Addison-Wesley Dix A., Finlay at all, Human-Computer Interaction, http://www.hcibook.com/e3/chaps/ch7/exercises/
4. Interaction Design, https://www.interaction-design.org/literature/topics/human-computer-interaction
5. Helen Sharp, Jennifer Preece, Yvonne Rogers, Interaction Design: Beyond Human-Computer Interaction Whiley, 2019

On-line resourses

6. URL http://www.e-learning.swu.bg

NORMS AND STANDARDS OF INFORMATION SECURITY

Semester: 6

Course Type: lectures

Hours (weekly)/WS/SS: 2 lectures per week / SS

ECTS Credits: 2 credits

Course Status: Optional course.

The development of e-business require secure infrastructure. Adopting a policy of compliance with world standards allows companies and organizations to implement best

practices. Information systems protection requires special regulations. Therefore criteria, standards, and in some cases, legislation on information security are set up. This ensure an adoption of best practices and adequate level of information security.

Course Objectives: Students gain knowledge and skills to cope with everyday and specific tasks related to the implementation of norms and standards related to information security. Get acquainted with the various policies and regulations for information security.

Teaching Methods: Lectures, demonstrations, work on project and teamwork.

Requirements/Prerequisites: Needed basic knowledge of operating systems, programming, computer architectures, computer networks and communications.

Assessment: written final exam

Course enrolment: Students should submit an application at the academic affairs department at the end of the current semester

References:

1. Mark Rhodes-Ousley, Information Security (Second Edition), The complete reference, McGraw-Hill, 2013
2. Brady Orand, Foundations of IT Service Management with ITIL 2011: ITIL Foundations Course in a Book, 2011
3. Edward Humphreys, Implementing the ISO/IEC 27001 Information Security Management System Standard, Artech House, 2007
4. Jule Hintzbergen, Foundations of Information Security Based on ISO27001 and ISO27002, Van Haren Publishing, 2010
5. Cybercrime Exposed, McAfee White paper

(http://www.mcafee.com/us/resources/white-papers/wp-cybercrime-exposed.pdf)

6. Сайт на Националният Център за Действие при Инциденти в Информационната Сигурност (https://govcert.bg/)
7. Нина Синягина, Иван Мирчев, Иво Дамянов, Светослав Христов, Защита на компютърната информация, УИ „Неофит Рилски“, 2005
8. https://www.pcisecuritystandards.org/security_standards/documents.php
9. http://www.itil-officialsite.com/

http://www.iso.org/iso/standards_development/processes_and_procedures/iso_iec_directives_and_iso_supplement.htm

WEB CONTENT MANAGEMENT

Semester: 6

Course Type: lectures and labs

Hours (weekly)/WS/SS: 1 lectures and 1 lab per week / SS

ECTS Credits: 2 credits

Course Status: Optional course.

Short Description: Modern ways of organizing and building of content on the web - blogs, wikis, social media requires more organized content management. Integration of web sites with features from social networks and building shared content through various Web services are important elements of the management of any modern website. This course will examine the important activities related to the promotion of content through indexing and search engine optimization. Presented are theoretical methods based on formal concept analysis, adaptive ontologies and programming for the organization of the structure and content sites. The course discusses specific systems for content management (CMS) - Sitecore, Umbraco, WordPress, Joomla and others.

Course Aims: To familiarize students with various activities tasks in content management and website building.

Teaching Methods: Lectures, Labs, Discussions, Project Based Methods

Requirements/Prerequisites: Knowledge in Programming Basics and HTML.

Exam: final exam

Course enrolment: the course is compulsory

Registration for the exam: Coordinated with lecturer and Students Service Department

References:

1. Денис Колисниченко, Да направим собствен сайт, Асеневци, 2015
2. Тим Киберман, На първо място в Google - 2015, Франчайзинг БГ, 2015
3. Michael Kuhlmann, Social Media for Wordpress, Packt Publishing, 2012
4. Hawker, Mark D., The developer's guide to social programming: building social context using Facebook, Google friend connect, and the Twitter API / Mark D. Hawker, Addison-Wesley, 2010
5. Dan Zarrella, The Social Media Marketing Book, O'Reilly Media, 2009
6. Brad Williams, David Damstra, Hal Stern, Professional WordPress, 3rd Edition, Wrox Publishing, 2015
7. John West, Professional Sitecore Development, Wrox Publishing, 2012
8. Alan Harris, Pro ASP.NET 4 CMS, Apress, 2010
9. Amanda Perran, Shane Perran, Jennifer Mason, Laura Rogers, Beginning SharePoint 2013, Wrox Publishing, 2013

Nik Wahlberg, Paul Sterling, Umbraco User's Guide, Wrox Publishing, 2011

LANGUAGE CULTURE

Semester: 6

Course Type: seminars exercises

Hours per week/SS: 2 hours seminars per week / SS

ECTS credits: 2 credits

Course Status: Optional Course.

COMPUTER NETWORKS AND COMMUNICATIONS

Semester: 7

Type of teaching: Lectures and laboratory work

Lessons per week: 2 lecture hours; 1 lab exercise hour per week

Number of credits: 6,5

Course status in the curriculum:

Compulsory for the students of speciality "Informatics" – bachelor degree.

Description of the course:

The course discuses the problems concerning design, building and application of computer networks. The lectures begin with introduction to computer networks, principles of building, historical development and their contemporary classification. Open system interconnection model of ISO is presented. Teaching course includes basic principles of building and functioning of Local Area Networks (LAN) illustrated by practical technical solutions in LAN Ethernet. The lectures on the most popular in the world computer network Internet present its basic characteristics, principles of functioning and application. The laboratory work helps to better rationalization of lecture material and contribute to formation of practical skills.

Aims and objectives of the course:

The aim of the course is to acquaint students with the basic principles, standards and tendencies of development in the field of computer networks. This will help them in future to professionally solve system tasks in the area of network communications.

Teaching methods:

Lectures (with slides, multimedia projector) and additional text materials; laboratory work (based on instructions) with a tutorial for every laboratory theme.

Prerequisites:

Basic knowledge in informatics.

Auxiliary means for teaching:

Computer and multimedia projector for the course. Computer, development software, local area network, Internet and a tutorial for every laboratory theme.

Method of assessment: written examination (work for fixed time).

Arrangement for examination: in the department office, coordinated with the lecturer.

ARTIFICIAL INTELLIGENCE

Semester: 7 semester

Course Type: lectures and seminar exercises

Hours per Week/FS/SS: 2 lecture hours and 1 seminar hour per week/FS

ECTS Credits: 6 credits

Course Status: Optional Course in the Informatics B.S. Curriculum

Course Description: The goals of Artificial Intelligence course are to present to students theoretical background in artificial intelligence field through common terminology, approaches and formalisms; to present functional and logical paradigms and programming and to present widely known and practically used methods and algorithms which are proved their acceptance in practice. A main part of this course is the knowledge presentation and elaboration of data in both functional and logical programming languages. In this course, functional and logical programming languages constructions are discussed. Main approaches are illustrated with a rich set of decisions of practical problems in seminar exercises. Classical directions of the field of artificial intelligence are discussed: search in the state space, knowledge presentation and usage of knowledge, human computer interaction by using restricted natural language, planning of actions, computer learning and knowledge extraction, image recognition.

Course Goals: The main goals of the course are: student should obtain knowledge and theoretical background about functional and logical programming and obtain practical skills in these two programming styles represented by Prolog and Scheme programming languages, respectively. Students will study classical notions and problems of artificial intelligence and some decisions and methods in artificial intelligence field.

Teaching Methods: The course uses classical forms of material presentation: lectures and seminars. The programming languages Prolog and Scheme will be used in practice for computer programming and problem decision descriptions.

Requirements/Prerequisites: Basic knowledge and experience in the following courses: Programming and Data Structures, Discrete Mathematics, Mathematical logic, etc.

Assessment: routine control 9usually 2 test-papers) and written final exam at the end of the semester.

Registration for the Course: by request at the end of the previous academic year

Registration for the Exam: coordinated with lecturer and Student Service Department

Abbreviation:

FS: Fall Semester

SS: Spring Semester

INTERNET PROGRAMMING

Semester: **7**

Type of Course: **Lectures and tutorials in computer lab.**

Hours per week: **2 hours lectures and 2 hours tutorials in computer lab/Fall semester.**

Credits Numbers: **6,5 credits**

Course Status: Core course in curriculum of major Informatics, Bachelor degree.

Course description:

The course is introduction in design and programming of Internet/Intranet Web-based information systems. Combination of HTML, Java-applets and MySQL/PHP/Apache technologies is considered in practical aspects.

Objectives: Basic objectives and tasks:

- The students give knowledge for algorithum thinking;
- to give knowledge for Data structures, that can process with computer;
- to give knowledge for methods and skills in programming;
- to give knowledge for good style in programming;
- to give knowledge for basic principles when develop applications;
- to give knowledge for design and programming of Internet/Intranet Web-based information systems;
- to give knowledge for practical aspects of HTML, Java-applets, and MySQL/PHP/Apache technologies.

Methods of teaching: lectures, tutorials, group seminars or workshop, projects, other methods

Pre-requirements: "Introduction to programming", "Object oriented programming", "Programming and data structures" and "Database systems".

Exam: Written examination and discussion at the end of the semester, individual tasks and the general student's work during the semester.

Registration for the Course: not necessary

Registration for the Exam: Coordinated with the lecturer and the Student Service Office

References:

1. Larry Ullman, PHP for the Web: Visual quickstart guide, Peachpit Press, 2016.
2. Денис Колисниченко, PHP 7 & MySQL - практическо програмиране, Асеневци, 2016
3. Саймън Стобърт, Дейвид Парсънс, Динамични уеб приложения с PHP и MySQL, DuoDesign, 2010

4. Мери Милхолън, Джеф Кастрина, Създаване на Web страници – бързо и ефективно. СофтПрес, 2014.
5. Нина В. Комолова, Елена С. Яковлева, HTML – самоучитель. Питер, 2011.
6. Сергей Соколов, CSS 3 в примери, Асеневци, 2009.
7. Adobe Dreamweaver CS – официален учебен курс. СофтПрес, 2008.
8. Дори Смит, Java за World Wide Web. ИнфоДАР, 2008.
9. BrianP. Hogan, HTML5 and CSS3. Develop with Tomorrow's Standards Today, Pragmatic Programmers, 2010
10. BrianP. Hogan, Web Design for Developers. A Programmer's Guide to Design Tools and Techniques, Pragmatic Programmers, 2010
11. Robin Nixon, Learning PHP, MySQL, and JavaScript, O'Reilly Media, 2009

APPLIED STATISTICS

Semester: 7

Course Type: lectures, tutorials and lab exercises

Hours per Week/FS/SS: 2 lecture hours, 1 tutorial hour and 1 lab hour per week/FS

ECTS Credits: 5.5 credits

Course Status: Elective course in curriculum of major Informatics. Bachelor degree**.**

Course Description: The course is an introduction to nonparametric statistics and probability as well as application of new IT to this area.

Course Objectives: Student should obtain knowledge:

- to apply methods of nonparametric statistics in practice;
- to realize concrete applications with tools of IT.

Teaching Methods: lectures, tutorials, lab exercises, discussions, project based method.

Requirements/Prerequisites: Probability and Statistics, Information Technology

Assessment and Evaluation:

- project: 30%
- final test: 70%.

The course is successfully completed with at least 65% of the full scores.

Registration for the Course: by request at the end of the previous academic year

Registration for the Exam: coordinated with lecturer and Student Service Department

Abbreviation:

FS: Fall Semester; SS: Spring Semester

EXPERT SYSTEMS

Semester: **7**

Type of Course: **lectures and tutorials in computer lab**

Hours per week: **2 lecture hours + 1 seminar hour + 1 lab exercise hour per week**

Credits: **5,5 credits**

Course Status: Elective course in curriculum of major Informatics. Bachelor degree.

Course description:

Expert systems are the most mature and widely used commercial application coming out of artificial intelligence. In an expert system, the computer applies heuristics and rules in a knowledge-specific domain to render advice or make recommenda tions, much like a human expert would. Expert systems have managed to achieve fairly high levels of performance in task areas which require a good deal of specialized knowledge and training. Often they perform tasks which are physically difficult, tedious, or expensive to have a human perform.

Objectives:

The student should obtain knowledge of:

- Basic principals of expert systems
- Knowledge bases
- Conceptual and semantic approaches of data modeling.

Methods of teaching: lectures, tutorials, discussions, project based method.

Pre- requirements: Functional Programming, Logical Programming, Artificial Intellect

Assessment and Evaluation

Quizzes - 20%

Final Test - 80%

The course is successful completed with at least 65% of all scores.

Registration for the Course: by request at the end of the current semester

Registration for the Exam: coordinated with the lecturer and the Student Service Office

References

1. Джаксън П. Увод в експертните системи, Софтех, София, 1999

2. Нишева, М., Д. Шишков. *Изкуствен интелект.* Издателство "Интеграл", Добрич, 1995.

3. Тодорова, М. *Езици за функционално и логическо програмиране,* първа част: функционално програмиране. София, Ciela, 200 v

4. Тодорова, М. *Езици за функционално и логическо програмиране,* втора част: логическо програмиране. София, Ciela, 2003.

NUMERICAL OPTIMIZATION

Semester: 7

Course Type: lectures and lab exercises

Hours per Week/FS/SS: 2 lecture hours and 2 lab hours per week/FS

ECTS Credits: 5.5 credits

Course Status: Optional Course in the Informatics B.S. Curriculum

Course Description: The course in Numerical Optimization includes basic numerical methods for solving various classes of optimization problems: line search methods – dichotomous search, golden section method, Fibonacci search, Newton's method; unconstrained optimization methods – nongradient methods (cyclic coordination method, method of Hooke and Jeeves, method of Rosenbrock), gradient methods (steepest descent method), methods of second order (Newton's method, modifications), as well as conjugate directions methods (conjugate gradients method: method of Fletcher-Reeves, method of Polak-Ribiere; quasi-Newton methods: method of Davidon-Fletcher-Powell); constrained optimization – methods of feasible directions (of Zoutendijk, of Rosen, of Wolfe [of the reduced gradient]), penalty and barrier functions methods; nonsmooth analysis and methods for nondifferentiable (nonsmooth) optimization; stochastic programming; separable programming; dynamic programming; vector (multi-objective) optimization and Pareto optimality. **Course Objectives:** Student should obtain knowledge and skills for numerical solution of optimization problems.

Teaching Methods: lectures and lab exercises

Requirements/Prerequisites: Basic knowledge in Mathematical analysis, Linear algebra, Analytic geometry, Mathematical programming.

Assessment: written final exam on two topics (grade weight is 60 %); two homework projects (grade weight is 40 %)

Registration for the Course: by request at the end of the previous academic year

Registration for the Exam: coordinated with lecturer and Student Service Department

References:

Basic Titles:

1. Yu. P. Zaichenko – "Operations Research", Slovo, Kiev, 2003 (in Russian).
2. V.G. Karmanov – "Mathematical Programming", 6-th ed., Fizmatlit, Moscow, 2008 (in Russian).
3. Stefan M. Stefanov – "Quantitative Methods of Management", 2003 (in Bulgarian).

Additional Titles:

4. M. S. Bazaraa, H. D. Sherali and C. M. Shetty – "Nonlinear Programming. Theory and Algorithms", John Wiley & Sons, Inc., New York, 3-rd ed., 2006.

5. R. Fletcher – "Practical Methods of Optimization", 2-nd ed., John Wiley & Sons, Chichester-New York-Brisbane-Toronto-Singapore, 2000.
6. Jorge Nocedal, Stephen Wright – "Numerical Optimization", 2-nd ed., Springer, 2006.
7. Stefan M. Stefanov – "Separable Programming. Theory and Methods", 4-th ed., Springer, Dordrech-Boston-London, 2016.

Abbreviation:

FS: Fall Semester

SS: Spring Semester

OBJECT-ORIENTED AND DISTRIBUTED DATABASES

Semester: 7

Course Type: lectures and tutorials in computer lab

Hours per week/SS: 2 hours lecture and 2 hours tutorials in computer lab / FS

ECTS credits: 5.5 credits

Course Status: Optional Course in Bachelor of Science Curriculum of Mathematics and Informatics

Course Description: This course is an introduction to methods for developing object-oriented and distributed databases by means of object-oriented environments for Visual design and event-oriented programming. It discusses the various aspects of the design of distributed databases and the use of various objects: datasets, objects of type "field" controls bound to data, etc. Developed various applications for data access depending on their architecture: object-oriented and distributed (client/application server/server). Study the various data access technologies such as ADO, dbExpress, IBExpress, ADO.NET, DataSnap, Cloud applications, etc.

Course Objectives This course aims to provide students with knowledge and additional training in the development of object-oriented and distributed databases and methods of their use.

Teaching Methods: Lectures, demonstrations, work on project and teamwork.

Requirements: Needed basic knowledge of databases (core course), database systems and object-oriented programming.

Assessment: Evaluating the student shall be carried out in the sixth grad scale – 2, 3, 4, 5, 6. Evaluation of current control is obtained by taking the average of the assessment of coursework and paper. Students who have a minimum average estimate /3/ of the current control is not allowed to test the regular session. They must present additional development and evaluation after receiving at least medium /3/ be admitted to the written examination of supplementary or liquidation session. The final estimate is derived from the average of the current control and evaluation of the written exam.

Registration for the Course: Submitted an application to the academic department at the end of current semester.

Registration for the Exam: Coordinated with the lecturer and Student Service Department.

References:

1. Embarcadero Technologies. Developing Database Applications: Embarcadero Technologies. Retrieved from Embarcadero Technologies Web Site. 2015.

2. Marco Cantu. Mastering Delphi 7. Publisher Sybex. 2003.
3. Marco Cantu. Delphi 2010 Handbook: A Guide to the New Features of Delphi 2010. 2010.
4. Mario Szpuszta, Ingo Rammer. Advanced .NET Remoting. Publisher: Apress; 2nd ed. 2005.
5. Bob Swart. Delphi XE DataSnap Development Essentials. Bob Swart Training & Consultancy. 2011.
6. Cary Jensen Ph,D. Delphi in Depth: ClientDataSets, Publisher CreateSpace Independent Publishing Platform. 2011.
7. Andrew Troelsen. Pro C# 5.0 and the .Net 4.5 Framework, Apress. 2012.
8. Tim Patrick. Microsoft ADO.NET 4 Step by Step. Publisher: Microsoft Press. 2010.
9. Xavier Pacheco. Delphi for .NET Developer's Guide. Publisher: Sams Publishing. 2004.

Abbreviation: FS: Fall Semester

MULTIMEDIA DATABASES

Semester: 7

Course Type: lectures, lab exercises

Hours per week/SS: 2 lecture hours per week and 2 labs hours per week/FS

ECTS credits: 5.5 credits

Course Status: Optional Course in Master of Science Curriculum of Informatics

Course Description: The course acquaints students with multimedia systems and multimedia databases, and various formats for multimedia data. Various elements of multimedia: the presentation of sound, text, images and graphics, animation and video. Studied data structures, models and multimedia data management systems databases. Internet is seen as a medium for distributed multimedia databases.

Course Objectives: The aim of the course is for students to gain an idea of basic concepts, objects and some basic theoretical results in the media, the ways in which multimedia objects are structured and can be described.

After completion of the course students should be able to:

- design effective and quality multimedia products.
- teamwork.

Teaching Methods: Lectures, demonstrations, work on project and teamwork.

Requirements: Needed basic knowledge of operating systems, computer programming and data structures, object-oriented programming, databases and DBMS. Desirable Knowledge of programming languages C + +, ObjectPascal and / or C #.

Assessment: Evaluating the student shall be carried out in the sixth grad scale – 2, 3, 4, 5, 6. Evaluation of current control is obtained by taking the average of the assessment of coursework and paper. Students who have a minimum average estimate /3/ of the current control is not allowed to test the regular session. They must present additional development and evaluation after receiving at least medium /3/ be admitted to the written examination of

supplementary or liquidation session. The final estimate is derived from the average of the current control and evaluation of the written exam.

Registration for the Course: Submitted an application to the academic department at the end of current semester.

Registration for the Exam: Coordinated with the lecturer and Student Service Department.

References:

Basic Titles:

1. Ian H. Witten, David Bainbridge, David M. Nichols (2009) *Multimedia Information and Systems: How to Build a Digital Library*, Morgan Kaufmann Publishing
2. Steve Heath (2017) *Multimedia and Communications Technology*, 2nd Edition, Focal Press
3. K. Selçuk Candan and Maria Luisa Sapino (2010) *Data Management for Multimedia Retrieval*, Cambridge University Press
4. Raphael Troncy, Benoit Huet, Simon Schenk (2011) *Multimedia Semantics: Metadata, Analysis and Interaction*, Publisher Wiley, USA
5. Marcelle Kratochvil (2013) *Managing Multimedia and Unstructured Data in the Oracle Database: A revolutionary approach to understanding, managing, and delivering digital objects, assets, and all types of data*, Packt Publishing, UK
6. Steve Heath (2017) Multimedia and Communications Technology, 2nd Edition, Focal Press.
7. Peter Brinkmann (2012) *Making Musical Apps: Real-time audio synthesis on Android and iOS*, O'Reilly Media, USA
8. Elizabeth Keathley (2014) *Digital Asset Management: Content Architectures, Project Management, and Creating Order out of Media Chaos*, Apress Publishing, UK
9. Matthew A. Russell (2013) *Mining the Social Web, 2nd Edition: Data Mining Facebook, Twitter, LinkedIn, Google+, GitHub, and More*, O'Reilly Media, English
10. Dan Zarrella (2009) The Social Media Marketing Book, O'Reilly Media, USA
11. Anton Polinger (2011) *Developing Microsoft Media Foundation Applications*, Microsoft Press, USA
12. Olive Marsh (2017) *Multimedia Technology and Applications*, ML Books International.

Abbreviation:

FS: Fall Semester

DECISION MAKING SUPPORT ALGORITHMS IN ECONOMICS AND MANAGEMENT

Semester: 7

Course Type: lectures and lab exercises

Hours per Week: 2 lecture hours and 2 lab hours per week/FS

ECTS Credits: 5.5 credits

Course Status: Optional Course in the Informatics B.S. Curriculum

Course Description: The course in Decision making support algorithms in economics and management includes four basic topics:

The first topic is dedicated to decision making methods by voting. Some basic voting algorithms and methods are considered;

The second topic includes basic methods and algorithms for solving of multi-objective (vector) problems;

The third topic is dedicated to the application of game theory in optimal decision making;

The fourth topic includes some methods and algorithms for decision making in risk conditions, and uncomplete information .

Course Objectives: Student should obtain knowledge and skills for some basic methods and algorithms supporting decision making.

Teaching Methods: lectures and lab exercises

Requirements/Prerequisites: Basic knowledge in Mathematical analysis, Linear algebra, Analytic geometry, Mathematical programming, Probability Theory.

Assessment: written final exam (grade weight is 60 %); two homework projects (grade weight 40 %)

Registration for the Course: by request at the end of the previous academic year

Registration for the Exam: coordinated with lecturer and Student Service Department

References:

1. V. A. Abchuk. "7:1 The Alphabet of the Solutions" – Tehnika, Sofia, 1986. (in Bulgarian)
2. T. R. Gichev, Z. K. Karamiteva. "Game Theory" – Nauka I Izkustvo , Sofia, 1980. (in Bulgarian)
3. G. H. Ivanov and coll. "Guide for Methematical Programing Problem Solving", UNI, Sofia, 1989 (in Bulgarian).
4. E. S. Venttzel. "Operation Research "– Nauka, Moscow, 1988 (in Russian).
5. Yu. I. Degtyarev. "Operation Research" – Higher School, Kiev, 1986 (in Russian).
6. Yu. K. Mashunin. "Vector Optimization – Methods and Models" – Nauka, Moscow, 1986 (in Russian).
7. Vira Chankong, Yacov Y. Haimes. Multiobjective Decision Making: Theory and Methodology Series Volume 8 – North-Holland, New York, Amsterdam, Oxford

MOBILE APPLICATION DEVELOPMENT

Semester: 7

Course Type: lectures and lab exercises

Hours per week/SS: 2 lecture hours and 2 lab hours per week / FS

ECTS credits: 5.5 credits

Course Status: Optional Course.

Course Description: Over the past few years have seen a rapid development of the market share of mobile devices such as tablets, e-readers and smartphones. Application development gained new meaning as the keyboard and mouse are no longer the main means of managing computing devices. Therefore, appears the need to learn new technologies and programming to create applications with a brand-new ideology. This course is a practical introduction to developing applications for mobile devices. In seminars, students will learn about the different environments to develop mobile applications and acquire basic theoretical knowledge and skills. Details will be discussed and used development environment Microsoft Visual Studio with Xamarin.Forms. It allows students to develop their applications in laboratory work and individual coursework at the end of the course.

Course Objectives This course aims to provide students with knowledge and additional training in the theory and practice in the development of applications for mobile devices. They will learn about some of the environments to develop of mobile applications and will gain more practical knowledge by Android application development with Xamarin.Forms.

Teaching Methods: Lectures, demonstrations, work on project and teamwork.

Requirements: Needed basic knowledge of operating systems, computer programming and data structures, object-oriented programming, databases and DBMS.

Assessment: Evaluating the student shall be carried out in the sixth grad scale – 2, 3, 4, 5, 6. Evaluation of current control is obtained by taking the average of the assessment of coursework and paper. Students who have a minimum average estimate /3/ of the current control is not allowed to test the regular session. They must present additional development and evaluation after receiving at least medium /3/ be admitted to the written examination of supplementary or liquidation session. The final estimate is derived from the average of the current control and evaluation of the written exam.

Registration for the Course: Submitted an application to the academic department at the end of current semester.

Registration for the Exam: Coordinated with the lecturer and Student Service Department.

References:

Basic Titles:

1. Paul Johnson (2018) Using MVVM Light with your Xamarin Apps, Apress.
2. Paul F. Johnson (2015) Cross-platform UI Development with Xamarin.Forms, Packt Publishing.
3. Janathan Peppers (2014) Xamarin Cross-platform Application Development, Packt

Publishing.

4. David Britch (2017) Enterprise Application Patterns using Xamarin.Forms, Microsoft Press.
5. Jim Bennett (2018) Xamarin in Action. Creating native cross-platform mobile apps, Manning Publications.
6. Russell Fustino (2018) Azure and Xamarin Forms: Cross Platform Mobile Development, Apress.
7. Charlez Petzold (2016) Creating Mobile Apps with Xamarin.Forms, Microsoft Press.
8. Matthew Leibowitz (2015) Xamarin Mobile Development for Android Cookbook, Packt Publishing.
9. Mark Reynolds (2014) Xamarin Essentials, Packt Publishing.
10. Dan Hermes (2015) Xamarin Mobile Application Development, Apress.
11. Can Bilgin (2016) Mastering Cross-Platform Development with Xamarin, Packt Publishing
12. Christopher Miller (2017) Cross-platform Localization for Native Mobile Apps with Xamarin, Apress.
13. William Smith (2014) Learning Xamarin Studio, Packt Publishing.
14. Mathieu Nayrolles (2015) Xamarin Studio for Android Programming: A C# Cookbook, Packt Publishing.
15. Jonathan Peppers (2014) Xamarin Cross-platform Application Development, Packt Publishing.
16. Michael Williams (2016) Xamarin Blueprints, Packt Publishing.
17. Cesar de la Torre, Simon Calvert (2016) Microsoft Platform and Tools for Mobile App Development, Microsoft Press.
18. Ayan Chatterjee (2017) Building Apps for the Universal Windows Platform, Apress.
19. Benjamin Perkins, Jacob Vibe Hammer, Jon D. Reid (2016) Beginning Visual C#® 2015 Programming, John Wiley & Sons, Inc.
20. Maximiliano Firtman (2013) Programming the Mobile Web, Second Edition, O'Reilly.
21. Gail Rahn Frederick, Rajesh Lal (2009) Beginning Smartphone Web Development, Apress.
22. Gerald Versulius (2017) Xamarin Continuous Integration and Delivery, Apress.
23. Adam Nathan (2016) Universal Windows® Apps with XAML and C#, SAMS
24. Xamarin.Forms Notes for Professionals; https://books.goalkicker.com/XamarinFormsBook/

Additional Titles:

1. Free ebook: Creating Mobile Apps with Xamarin.Forms; https://blogs.msdn.microsoft.com/microsoft_press/2016/03/31/free-ebook-creating-

mobile-apps-with-xamarin-forms/

2. Xamarin.Forms; https://docs.microsoft.com/en-us/xamarin/xamarin-forms/
3. Xamarin; https://docs.microsoft.com/en-us/xamarin
4. Microsoft Visual Studio; https://visualstudio.microsoft.com/

Abbreviation:

FS: Fall Semester

INTERACTIVE MULTIMEDIA TECHNOLOGIES

Semester: 7

Hours per week: 2 lecture+2 lab

ECTS credits: 5.5

Form of assessment: on-going control and exam

Examination type: practical task, solved on computer and computer based test

Annotation: The course is proposed for students from specialties "Informatics" and "Mathematics and informatics"". The main aim of the course is students to master basic methods and technics for design, development and integration of different multimedia objects.

Outcomes:

- Students have to be able to:
- Create, edit, and integrate different multimedia objects;
- Develop multimedia content;
- Design and develop interactie educational games

Discipline content: The course topics cover basic concepts of interactive multimedia, characteristics of authoring tools for development of interactive multimedia content. Also basic technologies for development of interatctive mobile applications and virtual reality are considered. The practical implementation is related to design and development of serious games.

Literature:

1. Иванов И. Интерактивни презентации, Изд. "Обучение", София, 2010
2. Иванов И. С. Николов, Цифрови видеопродукции, Изд. "Обучение", София, 2012
3. Марков А., М. Тодорова, М. Петров, Мултимедийни технологии, Фабер, Велико Търново, 2006
4. Тодорова М, Хр. Монева, "Мултимедийни технологии", УИ „Св. св. Кирил и Методий", Велико Търново, 2006 год. ,
5. Adobe Flash Professional CS6. Официален курс на Adobe Systems

6. Audacity Manual, http://manual.audacityteam.org/o/
7. Captivate 7 Maual, helpx.adobe.com/pdf/captivate_reference.pdf
8. Technical Support VideoPad Video Editor, http://www.nchsoftware.com/videopad/support.html
9. Минковска Д., МУЛТИМЕДИЯ И ВИРТУАЛНА РЕАЛНОСТ – ПРЕДИЗВИКАТЕЛСТВО ЗА НОВИТЕ ИНЖЕНЕРНИ ТЕХНОЛОГИИ http://www.tu-sofia.bg/faculties/mf/adp/nntk_files/konf-12/Materials/NAPRAVLENIE-8/10-8-D.Minkovska.pdf
10. Interactive Multimedia, Edited by Ioannis Deliyannis, ISBN 978-953-51-0224-3, 312 pages, Publisher: InTech, 2012, URL: http://www.intechopen.com/books/interactive-multimedia
11. Grigore Burdea, Philippe Coiffet: Virtual Reality Technology, Second Edition, John Wiley & Sons, 2003.

Technology of education and grading:

The focus of educational technology is grounded on problem based, project based and portfolio approaches. The students are supported by additional eLearning materials, published at www.e-learning.swu.bg .

Final grade consists of two grading's - on-going control and exam. On-going control is based on the student's portfolio. Exam has two parts: practical problem solving by computer and computer based test.

Grading scale:

A. Excellent 92% - 100%	D. Satisfied 50% - 58%
B. Very Good 76% - 91%	E. Fail 0% - 50%
C. Good 59% -75%	

NOSQL DATABASES

Semester: 7

Course Type: lectures and labs

Hours (weekly)/WS/SS: 2 lectures and 2 labs per week / WS

ECTS Credits: 5.5 credits

Course Status: Optional course.

Short Description: With the expansion of big data processing and information storage o non-relational databases become more popular. This course aims to introduce the basics of non-relational databases and to show how they can be used in specific projects. The course will examine three main types of non-relational databases - key-valued stores, document stores, column-oriented stores. Two NoSQL databases will be introduced in deep - MongoDB and RavenDB. Cloud storages as blobs, azure tables and dynamoDB will also be introduced.

Course Aims: Students should acquire knowledge and skills to work with non-relational databases.

Teaching Methods: Lectures, Labs, Discussions, Project Based Methods

Requirements/Prerequisites: Knowledge in Programming Basics, Object Oriented Programming, Operating Systems and Databases.

Exam: final exam

Course enrolment: Students should submit an application at the academic affairs department at the end of the current semester

Registration for the exam: Coordinated with lecturer and Students Service Department

References:

1. Gaurav Vaish, Getting Started with NoSQL, Packt Publishing, 2013
2. Dan Mccreary, Ann Kelly, Making Sense of NoSQL, Manning Publications, 2014
3. Shashank Tiwari, Professional NoSQL, Wrox, 2011
4. Christof Strauch, NoSQL Databases (http://www.christof-strauch.de/nosqldbs.pdf)
5. Eelco Plugge, Peter Membrey and Tim Hawkins, The Definitive Guide to MongoDB: The NoSQL Database for Cloud and Desktop Computing, Apress, 2010
6. David Chappell, Understanding NoSQL on Microsoft Azure, Chappell & Associates, 2014
7. http://ravendb.net/docs
8. https://foundationdb.com/documentation/
9. http://neo4j.com/developer/get-started/
10. http://hadoop.apache.org/docs/current/
11. Blob Service Concepts (https://msdn.microsoft.com/en-us/library/azure/dd179376.aspx)
12. http://cassandra.apache.org/

METADATA

Semester: 7

Course Type: lectures and labs

Hours (weekly)/WS/SS: 2 lectures and 2 labs per week / WS

ECTS Credits: 5.5 credits

Course Status: Optional course.

Short Description: The metadata can be used to facilitate the detection of the resources to anotate the contents of the database, and tracking the status of resources in a collection. Important for IT specialists is to assess the role of metadata in the: administration of digital

resources, security provision, data extraction and e-commerce. The course is designed to provide experience in applying metadata in practice. Covered in the course standards are Dublin Core Metadata, Resource Description Framework, and various types of microformat metadata. In addition to this will be introduced metadata in programming and their usage with Code Reflection.

Course Aims: Provide theoretical and practical background to students about the principles, standards and tools for working with metadata

Teaching Methods: Lectures, Labs, Discussions, Project Based Methods

Requirements/Prerequisites: Knowledge in Programming Basics, Object Oriented Programming, Operating Systems and Databases.

Exam: final exam

Course enrolment: Students should submit an application at the academic affairs department at the end of the current semester

Registration for the exam: Coordinated with lecturer and Students Service Department

References:

1. Liu, J. (2007). Metadata and its applications in the digital library: Approaches and practices. Westport, Conn: Libraries Unlimited.
2. Baca, M. (2008). Introduction to metadata. Online Edition, Los Angeles, CA: Getty Research Institute
3. (http://www.getty.edu/research/publications/electronic_publications/intrometadata/pdf.html)
4. Intner, S. S., Lazinger, S. S., & Weihs, J. R. (2006). Metadata and its impact on libraries. Library and information science text series. Westport, Conn: Libraries Unlimited.
5. http://dublincore.org/
6. Bert Moss, Metadata in Digital Forensics, eForensics Magazine, http://www.isebahamas.com/Bert%20Moss%20eforensics%20article.pdf
7. Usama Salama, Vijay Varadharajan, and Michael Hitchens, Metadata Based Forensic Analysis of Digital Information in the Web, Annual Symposium On Information Assurance & Secure Knowledge Management, June 5-6, 2012, Albany, NY (http://www.albany.edu/iasymposium/proceedings/2012/6-Salama_Varadharajan&Hitchens.pdf)
8. Дейвид Хънтър, Програмиране с XML, SoftPress, 2001
9. Shelley Powers, Practical RDF Paperback, O'Reilly, 2003

MSDN (http://msdn.microsoft.com)

XML STANDARDS FOR FILE FORMATS OF MS OFFICE

Semester: 7

Course Type: lectures and laboratory exercises

Hours per week FS/SS: 2 lecture hours and 2 laboratory hours per week /FS

ECTS credits: 5.5 credits

Course Status: Optional Course.

Course Description:

The course includes basics of the languages WordprocessingML, SpreadsheetML, PresentationML and DrawingML, and includes topics: introduction to Open XML, introduction to language WordprocessingML (Wordprocessing Markup Language), create electronic documents, creating a basic structure of a document WordprocessingML; wordprocessingML language capabilities for adding and formatting test, working with Tables in WordprocessingML; capabilities language WordprocessingML to work with styles, add images in WordprocessingML; capabilities language WordprocessingML formatting pages, adding custom XML text documents WordprocessingML, final layout of the document WordprocessingML; additional language options WordprocessingML; introduction to language SpreadsheetML (Spreadsheet Markup Language), key elements of a table (Spreadsheet), create worksheets (Worksheets) language SpreadsheetML; spreadsheetML options of language for working with formulas, optimizing worksheets, spreadsheetML options of language for working with spreadsheets and pivot tables; spreadsheetML language capabilities for adding and positioning diagrams, applying styles to content, conditional formatting, worksheet charts; additional language options SpreadsheetML; introduction to language PresentationML (Presentation Markup Language), creating a basic structure of the document in PresentationML; presentationML language capabilities for working with objects (Shapes), key elements of a presentation; capabilities language PresentationML to work with Placeholders and Images (Pictures); presentationML language capabilities for working with tables and charts; introduction to language DrawingML (Drawing Markup Language), drawingML capabilities of the language for text and graphics; drawingML language capabilities for working with tables and charts. Use themes;

Course Objectives:

Students should obtain fundamental knowledge and skills related to the basics of the languages WordprocessingML, SpreadsheetML, PresentationML and DrawingML.

Teaching Methods: lectures and laboratory exercises

Requirements/Prerequisites: Basic knowledge and skills for information systems and technology, as well as MS Word, MS Excel и MS PowerPoint, XML and HTML.

Assessment: written final exam

Registration for the Course: by request at the end of the previous academic year

Registration for the Exam: coordinated with lecturer and Student Service Department

References:

1. Wouter van Vugt. Open XML the Markup Explained. Microsoft Press. 2007.

2. Introduction to markup compatibility. https://msdn.microsoft.com/en-us/library/office/ ff478576.aspx. 2020.

3. Working with WordprocessingML documents. https://msdn.microsoft.com/EN-US/library/office/ gg278327.aspx. 2020.

4. Working with SpreadsheetML documents. https://msdn.microsoft.com/en-us/library/office/ gg278328.aspx. 2020.

5. Working with PresentationML documents. https://msdn.microsoft.com/en-us/library/office/ gg278318.aspx. 2020.

6. Drawing class. https://msdn.microsoft.com/en-us/library/office/documentformat.openxml.word-processing.drawing.aspx. 2020.

Abbreviation:

FS: Fall Semester

SS: Spring Semester

SOFTWARE ENGINEERING

Semester: **8**

Type of Course: **lectures and lab exercises**

Hours per week: **2 lecture hours and 2 lab exercise hours per week**

Credits Numbers: **7,0 credits**

Course Status: Basic course in curriculum of major Informatics. Bachelor degree.

Course Description: Software engineering associate with the development of software using well-defined scientific principles, methods, and procedures. The outcome of software engineering is an efficient and reliable software product. The result of software engineering is an effective and reliable software product. The innovations observed today are the result of well-designed and quality developed software products. This course is a theoretical and practical introduction to the management of software engineering. During the lectures, the students will become acquainted with the necessary theoretical material, and during the laboratory sessions, they will apply the acquired knowledge in practical projects.

Course Objectives This course aims to provide students with knowledge of basic theoretical concepts and practical approaches related to software engineering.

Teaching Methods: Lectures, demonstrations, work on project and teamwork.

Requirements: Needed basic knowledge of operating systems, computer programming and Data structures, object-oriented programming, databases and DBMS.

Assessment: Evaluating the student shall be carried out in the sixth grad scale – 2, 3, 4, 5, 6. Evaluation of current control is obtained by taking the average of the assessment of coursework and paper. Students who have a minimum average estimate /3/ of the current

control is not allowed to test the regular session. They must present additional development and evaluation after receiving at least medium /3/ be admitted to the written examination of supplementary or liquidation session. The final estimate is derived from the average of the current control and evaluation of the written exam.

Registration for the Course: Not necessary.

Registration for the Exam: Coordinated with the lecturer and Student Service Department.

References:

Basic Titles:

25. Capers Jones (2010) "Software Engineering Best Practices Lessons from Successful Projects in the Top Companies", McGraw-Hill Companies.
26. Rob Stephens (2015) "Beginning Software Engineering", Wrox.
27. John Dooley (2011) "Software Development and Professional Practice", Apress.
28. Henry H. Liu (2009) "Software Performance and Scalability. A Quantitative Approach", John Wiley & Sons, Inc.
29. Per Runeson, Martin Höst, Austen Rainer, Björn Regnell (2012) "Case Study Research in Software Engineering. Guidelines and Examples", John Wiley & Sons, Inc.
30. Stephen R. Schach (2011) "Object-Oriented and Classical Software Engineering", 8th Edition, McGraw-Hill Companies, Inc.
31. Coral Calero, Mario Piattini, Editors (2015) "Green in Software Engineering", Springer.
32. Sam Guckenheimer, Neno Loje (2012) "Agile Software Engineering with Visual Studio (Microsoft Windows Development Series)", 2nd Edition, Addison-Wesley
33. Caitlin Sadowski, Thomas Zimmermann, Editors (2019) "Rethinking Productivity in Software Engineering", Apress Open.
34. Josh Tyler (2015) "Building Great Software Engineering Teams", Apress.
35. Priyadarshi Tripathy, Kshirasagar Naik (2015) "Software evolution and maintenance: a practitioner's approach", John Wiley & Sons, Inc.
36. Olga Filipova, Rui Vilão (2018) "Software Development from A to Z: A Deep Dive into all the Roles Involved in the Creation of Software", Apress.
37. Douglas Bell (2005) "Software Engineering for Students: A Programming Approach", 4-th Edition, Addison-Wesley.
38. Simple Easy Learning (2018) "Software Engineering Tutorial: Absolute Beginners"; https://www.tutorialspoint.com/software_engineering/index.htm
39. Ronald J. Leach (2016) "Introduction to Software Engineering", 2nd Edition, CRC Press.
40. Susan Lincke (2015) "Security Planning: An Applied Approach", Springer.

Additional Titles:

5. Free ebook: Creating Mobile Apps with Xamarin.Forms; https://blogs.msdn.microsoft.com/microsoft_press/2016/03/31/free-ebook-creating-mobile-apps-with-xamarin-forms/
6. Xamarin.Forms; https://docs.microsoft.com/en-us/xamarin/xamarin-forms/
7. Xamarin; https://docs.microsoft.com/en-us/xamarin
8. Microsoft Visual Studio; https://visualstudio.microsoft.com/

COMPUTER MODEL IN SCIENCE

Semester: 8

Course Type: lectures and practical seminars

Hours (weekly): 2 lecture hours and 2 seminar hour per week (SS)

ECTS Credits: 6.5

Course Status: Optional course from Informatics B.C. Curriculum

Course Description:

The course "Computer model in science" could be used as a basic or optional one for all subjects aiming broad university preparation. The course is adapted and applied in teaching in the departments "Informatics", "Pedagogy of Teaching of Mathematics and Computer Science", "Pedagogy of Teaching of Physics and mathematics". To take that course it is enough for the student to have accomplished regular mathematics and science secondary school courses. Among the main objectives of the course (in times of information revolution and new information technologies) is the good and timely orientation and the acquisition of general mathematical and scientific knowledge by students - it is adapted first of all for non-physics students.

The course comprises of separate modules and mostly of computer experiments. For each PC-experiment with color computer animation, graphics, numerical results, explanations are provided. There are experiments in mechanics, thermodynamics and molecular physics, oscillations and waves, electricity and magnetism; optics, quantum physics, math, ecologies, biophysics, nonlinear economics, astrophysics, etc. The possibility in the course of independent work the change the parameters and observe the results of a computer experiment allows the student to investigate interactively each model in each topic. All that results in promotion of interest towards scientific knowledge and new information technologies application along with learning motivation. PC-models in Mathematical methods of investigation are used in every field of science and technology. Differential Equations are the foundations of the mathematical education of scientists and engineers.

(PC –models also use a model of exponential growth, comparison with discrete equations, series solutions; modeling examples including radioactive decay and time delay equation, integrating factor, series solution. nonlinear equation , separable equations, families of solutions, isoclines, the idea of a flow and connection vector fields, stability, phase-plane analysis; examples, including logistic equation and chemical kinetics. resonance, coupled first order systems, examples and PC-models of nonlinear dynamics, order and chaos, attractors. etc.)

Course Aims: The main goal is the students to master the instruments and methods of modeling in science.

Teaching Methods: lectures, tutorials, homeworks, tests, etc.

Requirements/Prerequisites: Calculus I and II, Linear Algebra and Analytical Geometry, Differential Equations.

Exam: tests, home works, final exam.

Registration for the course: optional course

Registration for the exam: Coordinated with lecturer and Students Service Department

References: 1/ Kirkpatrick/ Wheeler **.** Physics. A World View. 2-nd ed. 1995]; 2/ Stewart J. Calculus. III ed. (AUBG). 1996.). 3/ Фулър. Х. Р. Фулър, Р. Фулър. Физиката в живота на човека, изд. Наука и изкуство. С, 1988. 4/ Pontryagin L.S., Differential equations and applications, Moscow, 1988 (in Russian). 5/ Boss. Lectures in mathematics. Differential equations, Moscow, 2004 (in Russian). 6/ Elsgolc. L. Differential equations and variational calculus, Moscow, 2000 (in Russian) 8/ Diacu. An Introduction to Differential Equations: Order and Chaos, Freeman, 2000. 9/ Максимов, М., Г.Христакудис. Физика 10.клас, Булвест, С., 2000; 10/ Райчев, П., Кр. Иванов и др. Физика за 10 клас (ч. 2. Вълни и частици), Просвета, С., 1991г. 12/ Файнман, Р., Р. Лейтон, М.Сендс. Файнманови лекции по физика. 15/ **www.exponenta.ru** ; 16/ Multimedia course "**Open Physics**–I and II"; 17/ **WinSet**. M-I, 2003 etc.; 18/ **www.mathworld.wolfram.com.**

Abbreviations: FS: Fall Semester; SS: Spring Semester

PATTERN REGONITION

Semester: **8**

Form of the course: **lectures/exercises**

Hours (per week): **2 lecture hours + 2 lab exercises per week, Spring semester**

Credits: **6,5 credits**

Department: **Informatics, Faculty of Mathematics and Natural Sciences**, Southwestern University "Neofit Rilsky" – Blagoevgrad, phone 073 / 588 532

Status of the course in the educational plan:

The course is to be chosen in the educational plan of specialty Informatics

Description of the course:

This course covers subjects required as a background for IT professionals. This course expands techniques based on AI as well as new information technologies. The course covers main principles of the Pattern Recognition theory. Some of the topics included are: data representation, discovering basic signs, determining optimal decisions procedures (using different approaches and parameter evaluation).

This course is in compliance with similar courses in US and Russia.

Course Aims:

The course aim is to give students good basic theoretical knowledge and practical experience in pattern recognition. To become familiar with building mathematical models which they should use to solve different problems for classification simple objects.

Teaching Methods: lectures, seminars, discussions, practical work, and homework.

Requirements/Prerequisites:

Basic knowledge in Analytic Geometry, Linear Algebra, Analysis, Numerical Methods, Optimization. Advance knowledge in Discrete Mathematics, Graph Theory, Programming, Formal Languages and Grammars.

Exam: course project and final exam.

Registration for the course: a request is made by students at the end of the current semester.

Registration for exam: coordinated with the lecturer and Students Service Department.

References:

1. Ту Дж., Гонсалес, Принципы распознавния образов, Мир, Москва, 1978

2. Горелик А., В. Скрипкин, Методы распознавания, Высшая школа, Москва, 1989

3. Горелик А. Общая постановка задачи распознавания объектов и явлений, Кибернетика, 1980, №6

4. Айзерман М., Браверман, Розоноэр Л. Метод потенциальных функций в теории обучения машин, Москва, Наука, 1970

5. Журавлев Ю. Об алгебрическом подходе к решению задач распознавания или классификации, Проблемы кибернетики, М., Наука, 1978

6. Post E. Formal Reductions of the General Combinatorial Decision Problem, Am J. of Math, 1943, v.65

7. Fu K. Syntacic (Linnguistic) Pattern Recognition.- to book "Digital Pattern Recognition", Edited by K. Fu, Second Corrected and Updated Edition, Springer - Verlag: Berlin-Heidelberg - New York,1980

8. Михалевич В., Последовательные алгориты оптимизации и их применение, Кибернетика, 1965, № 1 и 2

9. Looney C.G., Pattern Recognition using neural networks: theory and algorithms for engineers, Oxford University Press, 1997

INTERNET TECHNOLOGIES

Semester: **8**

Type of Course: **lectures and tutorials in computer lab.**

Hours per week – **2 lecture hours and 2 lab exercise hour per week**

Credits Numbers: **6,5 credits**

Course Status: Elective course in curriculum of major Informatics. Bachelor degree.

Course description:

The course is introduction in design of Web-based Internet/Intranet information systems based on Oracle Application Express technology.

Objectives:

The student should obtain knowledge of:

- Design of Internet/Intranet Web-based information systems.
- Practical aspects of Internet/Intranet Web-based information systems development.

Methods of teaching: lectures, tutorials, discussions, project based method.

Pre- requirements: Database systems (core course), Internet Programming (core course)

Assessment and Evaluation

Project- 50%

Final Test- 50%

The course is successful completed with at least 51% of all scores.

Registration for the Course: by request at the end of the current semester

Registration for the Exam: coordinated with the lecturer and the Student Service Office

References:

1. Oracle Corporation, Oracle Database® Database Concepts 19c, February 2020
2. Oracle Corporation, Oracle® Application Express Application Builder User's Guide, Release 19.2, December 2019
3. Oracle Corporation, Oracle® Application Express SQL Workshop Guide, Release 19.2, December 2019
4. Oracle Corporation, Oracle® Application Express Administration Guide, Release 19.2, October 2019
5. Rick Greenwald, Beginning Oracle® Application Express, ISBN 9780470388372.

COMPUTER SECURITY

Semester: 8

Course Type: lectures and lab exercises

Hours per Week/FS/SS: 2 lecture hours and 2 lab hour per week/SS

ECTS Credits: 6.5 credits

Course Status: Optional Course in the Informatics B.S. Curriculum

Course Description: In the course students are introduced in present risks connected with information storage and methods for its protection from damaging and unauthorized access. Beside theoretical aspects of its protection (codes, cryptographic schemes), different tools and methods for control and protection are observed. In the course, basic principles of trusted

systems, classifications and standards for estimation of trust on different classes of computer systems with respect to their protection are observed.

Course Aims: This course aim is to give students good basic knowledge in identification of possible risks and background in applying different methods of information protection.

Teaching Methods: lectures and lab exercises

Requirements/Prerequisites: Basic knowledge in Operating Systems, Computer Architectures, Probability and Statistics, Discrete Mathematics. Knowledge in Coding Theory and Cryptography is advantage.

Assessment: written final exam

Registration for the Course: by request at the end of the previous academic year

Registration for the Exam: coordinated with lecturer and Student Service Department

References:

Basic Titles:

1. Русин Ал. Петров – „Защита на информацията в компютрите и мрежите", ИК „Корени", 2003.

2. Matt Bishop, "Computer Security: Art and Science", Addison Wesley Professional, 2003.

3. Debby Russell, Sr. G.T. Gargemi, "Computer Security Basics", O'Reilly, 2003.

4. Joel Scambray, Stuart McClure, George Kurtz, "Hacking Exposed", Addison Wesley, 2003.

5. "Introduction to Cryptography", Network Associates, Inc. and its Affiliated Companies, 1998.

6. B. Dunsmore, J. Ballew, M. Cross, J. Harper, "Mission Critical: Internet Security", Syngress Press, 2001.

Abbreviation: FS: Fall Semester SS: Spring Semester

COMPUTER DESIGN

Semester: 8

Course Type: lectures and lab exercises

Hours per week/FS/SS: 2 lecture hours, 2 lab exercise hour per week/FS

ECTS credits: 6,5

Course Status: Optional course in the Computer Science BSc curriculum

Course Description: The course aims to introduce students to the theoretical foundations of graphic design and its importance to information technology. Software for creating and editing raster and vector images are used to illustrate the studied theory. The knowledge that students will gain will help them create and edit various graphic objects, create graphic galleries, know the rules for good design, and can select appropriate colours and fonts. The

course is adapted towards the students of the "Informatics" specialty, at South-West University "Neofit Rilski" Blagoevgrad. It is a continuation of the courses in Graphic Design of Printed and Promotional Materials, Mathematical Foundations of Computer Graphics, and Mobile Applications Development.

Course Objectives This course aims to provide students with knowledge and additional training in the theory and practice of graphic design. They will learn about the methods of digital image processing, how to create vector and raster graphics and animation.

Teaching Methods: Lectures, demonstrations, work on project and teamwork.

Requirements: Needed basic knowledge of information technology.

Assessment: Evaluating the student shall be carried out in the sixth grad scale – 2, 3, 4, 5, 6. Evaluation of current control is obtained by taking the average of the assessment of coursework and paper. Students who have a minimum average estimate /3/ of the current control is not allowed to test the regular session. They must present additional development and evaluation after receiving at least medium /3/ be admitted to the written examination of supplementary or liquidation session. The final estimate is derived from the average of the current control and evaluation of the written exam.

Registration for the Course: Submitted an application to the academic department at the end of current semester.

Registration for the Exam: Coordinated with the lecturer and Student Service Department.

References:

1. M. Monteiro (2012) "Design is a job", A Book Apart
2. P. Whitt (2016) "Pro Photo Colorizing with GIMP", Apress.
3. J. M. Ferreyra (2011) "GIMP 2.6 Cookbook", Packt Publishing.
4. T. Bah (2017) Inkscape: Guide to a Vector Drawing Program, 5th Edition; http://tavmjong.free.fr/INKSCAPE/MANUAL/html/.
5. B. Hiitola (2016) "Inkscape 0.48 Starter", Packt Publishing.
6. M. Jurkovic R. Di Scala (2011) "Inkscape 0.48 Illustrator's Cookbook", Packt Publishing.
7. W. Jackson (2015) "Digital Illustration Fundamentals", Apress.
8. L. Mathis (2016) "Designed for Use", 2nd Edition, Pragmatic Programmers.
9. J. Shariat, and C. S. Saucier (2017) "Tragic Design", O'Reilly Media.
10. D. Walsh (2015) "2D Game Art", AtomicVertex.com.
11. J. DiMarco (2010) "Digital Design for Print and Web", Wiley.
12. N. Iliinky, J. Steele (2011) "Designing Data Visualizations", O'Reilly Media.
13. J. Jatz (2012) "Designing Information. Human factors and common sense in information design", Wiley.
14. P. Shirley, S. Marschner (2009) "Fundamentals of Computer Graphics", CRC Press.
15. К. Уилкинсън (редактор) (2014) „Знаци и символи. Илюстрован справочник за техния произход и значение", Книгомания.

16. Ст. Малешков, В. Георгиев (2014) „Компютърна графика и фотореалистична визуализация“, Нов български университет.
17. В. Гличка (2016) Основи на векторната графика, Алекс Софт.
18. J. M. Blain (2016). The Complete Guide to Blender Graphics: Computer Modeling & Animation. AK Peters/CRC Press.
19. L. Flavell (2011). Beginning Blender: Open Source 3D Modeling, Animation, and Game Design. Apress.

Abbreviation:

FS: Fall Semester

SS: Spring Semester

DEVELOPING DATABASE APPLICATIONS

Semester: 8

Course Type: lectures, lab exercises

Hours per week/SS: 1 lecture hour per week and 3 labs hours per week/SS

ECTS credits: 6.5 credits

Course Status: Optional Course in Bachelor of Science Curriculum of Informatics

Course Description:

The course includes basic methods in database application development and related topics: designing applications for working with databases; use of data sets; use of data fields; working with standard controls on data; working with advanced controls for data; working with unidirectional datasets, technology overview dbExpress; using client datasets; create applications to work with databases using technology ADO; create applications to work with databases using technology BDE; methods for transforming existing BDE applications to applications based on newer technologies; using XML in applications for working with databases; export information from database applications to CSV and HTML formats; export information from database applications to MS Excel; analyzing information in applications for working with databases; creating reports in applications for working with databases.

Course Objectives:

Students should obtain basic knowledge and skills for developing databases applications.

Teaching Methods: lectures and laboratory exercises

Requirements/Prerequisites: Basic knowledge and skills for databases, database management systems and programming.

Assessment: written final exam

Registration for the Course: by request at the end of the previous academic year

Registration for the Exam: coordinated with lecturer and Student Service Department

References:

1. Embarcadero Technologies. (2020). Developing Database Applications: Embarcadero Technologies. Retrieved from Embarcadero Technologies Web Site:

2. Marco Cantu. (2003). Mastering Delphi 7. Publisher Sybex

3. Marco Cantu. (2005). Mastering Borland Delphi 2005. Publisher Sybex

4. Marco Cantu. (2010). Delphi 2010 Handbook: A Guide to the New Features of Delphi

5. Embarcadero Technologies. (2020). FireDAC: Embarcadero Technologies. Retrieved from Embarcadero Technologies Web Site:

6. Андрей Сорокин. (2005). Delphi разработка баз данных. Издателство: Питер.

7. Eric Harmon. (2001). Delphi/Kylix Database Development. Publisher Sams

8. Ivan Hladni. (2006). Inside Delphi. Publisher Wordware Publishing

9. David M. Kroenke, David Auer. (2012). Database Concepts (6th Edition). Publisher Prentice Hall, USA

10. Carlos Coronel, Steven Morris, Peter Rob. (2012). Database Systems: Design, Implementation, and Management. Publisher Cengage Learning, USA

Abbreviation:

SS: Spring Semester

E-TRAIDING AND CORPORATE INFORMATION SYSTEMS

Semester: 8

Type of Course: lectures and tutorials in computer lab.

Hours per week: 2 hours lectures and 2 hours tutorials in computer lab.

ECTS Credits: 6,5 credits

Course Status: Elective course.

Course description: The course is an introduction to e-business and corporate information systems.

Objectives:

The student should obtain knowledge of:

- Design of Internet/Intranet Web-based information systems.
- Practical aspects of Internet/Intranet Web-based information systems development.

Methods of teaching: lectures, tutorials, discussions, project based method.

Pre- requirements: Database systems (core course), Internet Programming (core course)

Assessment and Evaluation

Project- 50%

Final Test- 50%

The course is successful completed with at least 51% of all scores.

Registration for the Course: by request at the end of the current semester

Registration for the Exam: coordinated with the lecturer and the Student Service Office

References:

1. Turban, E., Volonino, L., Wood G., Information Technology for Management, 9th ed., Wiley, 2013
2. Ralph M. Stair, George W. Reynolds, Fundamentals of Information Systems, 6th ed., Course Technology, Cengage Learning, 2012
3. Димитров, Г., Право на информационните и комуникационните технологии, Фондация „Право и интернет", София, 2014

Williams, B., Sawyer, S., Using Information Technology: A Practical Introduction to Computers & Communication: Complete Version, 9th ed., McGraw Hill, 2011

INFORMATION RETRIEVAL AND WEB SEARCH

Semester: 8

Course Type: lectures and labs

Hours (weekly)/WS/SS: 2 lectures and 1 lab per week / SS

ECTS Credits: 6,5 credits

Course Status: Optional course.

Short Description: This course provide summary on databases, NoSQL databases, metadata, and through skillful application of domain specific languages to consider actions for data analysis and data mining. Discussed are topics related to: processing of large data storage, search on the web, indexing information using Solr, crawl and collect information on the web. The course will also address developments of search engines. Additionally, will be discussed options of knowledge extraction with ontologies and algorithms for patterns recognition.

Course Aims: Provide specialized training for analyzing data, generation of indexes, creation of modules for site search and tools to crawl and collect information

Teaching Methods: Lectures, Labs, Discussions, Project Based Methods

Requirements/Prerequisites: Knowledge in Programming Basics, Object Oriented Programming, Operating Systems and Databases.

Exam: final exam

Course enrolment: Students should submit an application at the academic affairs department at the end of the current semester

Registration for the exam: Coordinated with lecturer and Students Service Department

References:

1. Stefano Ceri at al., Web Information Retrieval, Springer-Verlag, 2013

2. Peter Morville and Jeffery Callender, Search Patterns, O'Reilly Media, Inc.,2010
3. Rafal Kuc, Apache Solr 4 Cookbook, Packt Publishing, 2013
4. J. Cho, N. Shivakumar, H. Garcia-Molina. Finding replicated web collections. ACM SIGMOD Record, Vol. 29, No. 2, pp. 355-366, 2000.
5. S. Chakrabarti, B. Dom, D. Gibson, J. Kleinberg, S.R. Kumar, P. Raghavan, S. Rajagopalan, A. Tomkins, Hypersearching the Web. Scientific American, June 1999.
6. S. Brin, L. Page. The anatomy of a large-scale hypertextual Web search engine. Computer Networks and ISDN Systems. Vol. 30, No. 1-7, pp. 107-117, 1998.

METHODS AND TOOLS FOR COMPUTER SYSTEMS INTEGRATION

Semester: 8

Course Type: lectures and labs

Hours (weekly)/WS/SS: 2 lectures and 2 lab per week / SS

ECTS Credits: 6**.5** credits

Course Status: Optional course.

Short Description: Modern development of information services for organizations suggests gradual development of information systems and their integration. The course offers an overview of the different architectures - based on common parts and service-oriented. Discussed are types of systems in the IT infrastructure of the organization such as passive and active systems, homogeneous and heterogeneous systems. The main topics are related to methods for systems integration: through common databases, messages and Web services.

Course Aims: to extend training of IT specialists in the integration and administration of various information computer systems.

Teaching Methods: Lectures, Labs, Discussions, Project Based Methods

Requirements/Prerequisites: Knowledge in Programming Basics, Object Oriented Programming, Databases, Operating Systems, Networking, Internet Programming

Exam: final exam

Course enrolment: Students should submit an application at the academic affairs department at the end of the current semester

Registration for the exam: Coordinated with lecturer and Students Service Department

References:

1. Microsoft, Pattern and Practice, Integration Patterns, 2004 (http://download.microsoft.com/download/a/c/f/acf079ca-670e-4942-8a53-e587a0959d75/intpatt.pdf)
2. Marcin Kawalerowicz, Craig Berntson, Continuous Integration in .NET, Manning, 2011
3. Carl Jones, Do More with SOA Integration, Packt Publishing, 2011

4. http://www.enterpriseintegrationpatterns.com/
5. IBM, Application Integration patterns 2004 (http://www.redbooks.ibm.com/redpapers/pdfs/redp3837.pdf)
6. SalesForce, Integration Patterns and Practices, 2015 (http://www.salesforce.com/us/developer/docs/integration_patterns/integration_patterns_and_practices.pdf)
7. Ben Morris Blog, (http://www.ben-morris.com/)