BSc Study program: **COMPUTER AND SOFTWARE ENGINEERING**

Overview of Courses in Study Program

1st year: Computer ans Software Engineering						
1st semester						
No	**M/E**	**Course**	**L**	**NE**	**LE**	**ECTS**
1	M	Linear Algebra with Calculus 1	3	3	0	7
2	M	Physics 1	3	1	1	6
3	M	Fundamentals of electrical engineering 1	3	1	1	7
4	M	Fundamentals of programming	2	0	2	5
5. *Select one of the following elective courses*						
5	E	Technical English	2	1	0	5
5	E	Communication skills	2	1	0	5
5	E	German Language	2	1	0	5
5	E	Practical Mathematics	2	1	0	5
2nd semester						
No	**M/E**	**Course**	**L**	**NE**	**LE**	**ECTS**
1	M	Calculus 2	3	3	0	7
2	M	Physics 2	3	1	1	6
3	M	Fundamentals of electrical engineering 2	3	1	1	7
4	M	Algorithms and Data Structures	2	0	2	5
5	M	Digital logic circuits	2	1	1	5

2nd year: Computer ans Software Engineering						
3rd semester						
No	**M/E**	**Course**	**L**	**NE**	**LE**	**ECTS**
1	M	Discrete Mathematics and Probability	2	2	0	5
2	M	Databases	2	0	2	5
3	M	Object Oriented Programming	2	0	2	5
4	M	Computer Architecture	2	0	2	5
5	M	Electronics	2	1	1	5
6	M	Web Programming I	2	0	2	5
4rth semester						
No	**M/E**	**Course**	**L**	**NE**	**LE**	**ECTS**
1	M	Operating Systems	2	0	2	5
2	M	Software Engineering	2	0	2	5
3	M	Data Security	2	0	2	5
4	M	Human Computer Interaction	2	0	2	5
5	M	Web Programming II	2	0	2	5
6. *Select one of the following non-technical electives (non-technical electives)*						
	E	Legal, Ethical and Social Issues in ICT	2	0	2	5
	E	Budget and cost analysis	2	0	2	5

3rd year: Computer ans Software Engineering						
5th semester						
No	**M/E**	**Course**	**L**	**NE**	**LE**	**ECTS**
1	M	Microprocessors and Microcontrollers	2	0	2	5
2	M	Design and Analysis of Algorithms	2	0	2	5
3	M	Computer Networks	2	0	2	5
4	M	Mobile devices programming	1	0	3	5
5. & 6. *Select two of the following technical electives (technical electives)*						
	E	Data Engineering	2	0	2	5
	E	Computer Security	2	0	2	5
	E	Software Testing	2	0	2	5
	E	Concurrent Computing	2	0	2	5
	E	Entrepreneurship and innovation	2	0	2	5
	E	IT project management	2	0	2	5
6th semester						
No	**M/E**	**Course**	**L**	**NE**	**LE**	**ECTS**
1	M	Distributed systems	2	0	2	5
2.&3. &4. *Select three of the following technical electives (technical electives)*						
	E	Data Mining	2	0	2	5
	E	Internet Security	1	0	2	5
	E	Big data	2	0	2	5
	E	Extraction of information	2	0	2	5
	E	Visual computing	2	0	2	5
	E	Cloud computing	2	0	2	5
	E	Parallel computing	2	0	2	5
	E	Data communication	2	0	2	5
	E	Biomedical Engineering	2	0	2	5
	E	Biometrics & Forensics	2	0	2	5
5	M	Professional practice				5
6	M	Diploma Thesis				5

Note: M- Mandatory, E- Elective, L- Lectures, NE- Numerical exercises, LE-Laboratory exercises

This study program is based on IEEE and ACM recommendations for the Computer Engineering study program, see link:

https://ieeecs-media.computer.org/assets/pdf/ce2016-final-report.pdf

Whereas, from the universities of the region we have taken as an example:

1. University of Zagreb
   - https://www.fer.unizg.hr/en/study_programs/undergraduate_study/computing, similarity 70-80%
2. Technical University of Vienna
   - https://tiss.tuwien.ac.at/curriculum/public/curriculum.xhtml?dswid=7221&dsrid=430&key=46100, similarity 70-80%

Short course descriptions

Course title: Linear Algebra with Calculus I

Lecturer: Prof. Asoc. Dr. Qefsere Gjonbalaj, Prof. Asoc. Dr. Shqipe Lohaj, Prof. Asoc. Dr. Valdete Rexhëbeqaj Hamiti
Course status: Mandatory, Semester I, 7 ECTS

Short course description: In this course will be studied: complex numbers, parts from linear algebra, analytic geometry and functions with one-variable.

Course objectives: Students should be trained so that the knowledge gained through this course can be applied as an ancillary device in electrical and computer studies.

Learning outcomes:
Upon completion of this course the student will be able to:

1. describe, solve and design various problems in the field of his profession when dealing with complex number operations, through matrices and determinants;
2. describe and solve problems related to systems of linear equations;
3. find the functional connections of the magnitudes of various electrical problems and then with differential calculations, describe and examine those functional connections;
4. understand the concept of the derivative and is able to apply it to many problems in Geometry, Electronics, Telecommunication, Informatics and other areas;
5. demonstrate skills of mathematical modelling and problem solving.

Teaching methodology: Lectures, discussions, exercises, consultations, homework, seminar work, intermediate evaluations, final exam.

Evaluation methods: Seminar work and homework (10%), first intermediate evaluation (25%), second intermediate evaluation (25%), final exam (40%).

Concretization tools/IT: Pencil, whiteboard, projector and computer.

Ratio between theoretical and practical part: Ratio between the theoretical and practical part is 2:1.

Literature:

- Hamiti E. - Matematika I, Prishtinë 2008.
- Hamiti E. - Matematika II, Prishtinë 2008.
- Peci H, Doko M. - Përmbledhje detyrash të zgjidhura nga Matematika I, Prishtinë 1997.
- Loshaj Z. – Përmbledhje detyrash të zgjidhura nga Matematika II, Prishtinë 1996.

Course title: Physics I

Lecturer: Dr.sc. Valon Veliu
Course status: Mandatory, Semester I, 6 ECTS

Short course description: The course includes basic knowledge of physics necessary to gain general knowledge that is basic in engineering.

Course objectives: Using the physical laws to solve the basic problems of engineering.

Learning outcomes: On successful completion of the course, students will be able to:

- Analyze simple mechanical systems and solve equations of motion.
- Apply principles of conservation of energy and momentum to particle collisions.
- Explain the conditions of statics of rigid bodies and the equation of motion for rotation of the rigid body around fixed axis.
- Understand the first law of thermodynamics and analyze thermodynamic cycles.

Teaching methodology: 45 hours of lectures + 15 hours of auditorial exercises + 15 hours of laboratory exercises. Approximately 75 hours of personal study.

Evaluation methods: Seminar 10%, Mid-term exams 30 %, Final Exam 60 %.

Concretization tools/IT: Pencil, whiteboard, projector and computer.

Literature:

- S. Skenderi, R,Maliqi, "Physic for thechnical faculty", UP, Pristina, 2005.
- J. Serway, Physics for scientists and engineering, Thomson Books, 2004.
- D. Haliday, R.Resnick, J.Walker, Fundametals of Physics, John Wiley &Sons, 2001

Course title: Fundamentals of electrical engineering 1

Lecturer: Prof. Dr. Luan Ahma, Prof. Dr. Enver Hamiti, Prof. Dr. Mimoza Ibrani, Prof. Ass. Dr. Vjosa Shatri

Course status: Mandatory, Semester I, 7 ECTS

Short Course description: Basics of electricity. Fundamental lows of electricity. Electric potential. An electric dipole and flux lines. Electrostatic induction. Electric generator. Polarization in dielectrics. Boundary conditions. Capacitance. Electrostatic networks. Energy and forces in electrostatic fields. Electrostatic field analysis using MATLAB. Concepts, elements and topology of electric circuits. Types of electrical circuits. Elementary DC circuits. Circuit Elements, Electrical resistance- Ohm's law. Construction of circuit model.Voltage and potential in electric circuit. Current source. Kirchoff's laws. Analysis of a Circuit Containing Dependent Sources. Complex DC circuits. DC circuit analysis using SPICE.

Course objectives: The purpose of the course is to introduce the basic principles of electrical field and DC current circuits.

Learning outcomes: On successful completion of the course, students will be able to:

1. Understand fundamental laws of electricity.
2. Apply the fundamental laws of electricity for solving of electric field problems.
3. Apply MATLAB software pavckage for soving basic problems in elctrical field.
4. Understand and apply methods for DC circuit analysis such as: Kirchhof's lows, node
voltage method, mesh current method, superposition method, Thevenin's and Norton's theorem.
5. Understand transient response of first order circuits (series RC circuits).
6. Apply PSPICE Software for DC circuit analysis.
7. Apply gained knowledge in other electrical engineering fields.

Teaching Methodology: Presentations, tutorials, discussions, and laboratory works meaning: 60 hours presentations + 15 hours of tutorials, and 15 hours laboratory works.

Evaluation: First assesment:30%, Second assesment: 25%, Home exercises 10%, Attendance: 5%, Final exam, 30%, Total:100%.

Concretization tools: Compjuter, video projector, and equiped lab with neccessary devices to ilustrate all teaching material.

Literature:

- Nexhat Orana, Bazat e elektroteknikës 1, Prishtinë, 1994
- M.N. Sadiku, Elements of electromagnetic, Oxford University Press, New York, 2001
- Ch. Alexander, M.N. Sadiku, Electric Circuits, McGraw Hill, New York, 2000.

Course title: Fundamentals of Programming

Lecturer: Prof.Ass. Dr. Avni Rexhepi, Prof. Asoc. Dr.Kadri Sylejmani
Course status: Mandatory, Semester I, 5 ECTS

The goal: The purpose of the course is to introduce the basic principles of programming and algorithms, for solution of problems with computer and writing of respective programs (in C++ programming language).

Learning outcomes: On successful completion of the course, students will be able to write programs in C++ programming language, including the use of different functions.

Course content: Introduction to algorithms: sum, product and factorial. Array operations and algorithms: sum/product of the array members, counting members, searching and sorting. Matrix manipulations: creation, sum/product of the members, searching, creating vector from the matrix members. Introduction to C++ programming language: variable types, operators, statements. Input and output: input of numbers, arrays, strings and their output, use of manipulators. Branching: if, goto, switch( ). Loops: while, do-while, for. Loops: breaking, continuing and exiting. Functions: defining and executing, parameters and arguments, mathematical and string functions. Using vectors and matrices with functions. Recursion, pointers, references. User defined types: enumerations, structures, classes.

Methods of teaching: 30 hours of lectures + 30 hours of auditory/lab exercises. Approximately 65 hours of personal study and exercise including home problems/tasks.

Grading System: Attendance lectures 10%, Attendance lab. 10%, Mid-term problems 30 %, Collocui/Final Exam 50 %
Literature:

- Agni Dika, "Algoritmet, me programe në C++", Universiteti i Prishtinës, Fakulteti Elektroteknik, Prishtinë, 2004, http://www.agnidika.net/algoritmetCpp.pdf
- Agni Dika Bazat e Programimit në C++, Universiteti i Europës Juglindore, Tetovë, ISBN 9989-866-23-6, http://www.agnidika.net/programimiCpp.pdf
- H.M. Deitel, P. J. Deitel, How to Program C++, Prentice Hall, Upper Saddle River, New Jersey, ISBN 0-13-111881-1.
- Robert Lafore, Object-Oriented Programming in C++, Sams, Indianopolis, Indiana, ISBN-10:0-672-32308-7.

Course title: Technical English

Lecturer: Lecturer from UP
Course status: Elective, Semester I, 5 ECTS

Short description: The focus of this course is to teach students to use English to clearly communicate and write engineering technical concepts.

Course goals: Enabling student to actively use the English language to communicate, orally and in writing, at the necessary level for the professional field of electrical and computer engineering.

Expected learning outcomes: Upon completion of this course the student will be able to:

1. actively use English in everyday life.
2. communicate both orally and in writing the English language at a desired level, specifically as it pertains to their professional field.
3. ask and answer questions in the field of electrical engineering in English.
4. translate texts related to the field of electrical and computer engineering.

Teaching methodology: 30 hours of lectures, 15 hours of exercises. Approximately 80 hours of independent work, including a seminar.

Assessment: Seminar 10%, Intermediary Evaluations 30%, Final Examination 70%.

Ratio between theoretical and practical parts of studying: 2:1

Primary literature:

- Day, Jeremy, *Cambridge English for Engineering*, Cambridge, UK

Course title: Communication skills

Lecturer: Prof. Dr. Blerim Rexha, Prof. Dr. Sabrije Osmanaj
Course status: Elective, Semester I, 5 ECTS

The goal: The goal of this course is to develop written and verbal communication skills and group work.

Learning outcomes: On successful completion of the course, students will be able to:

1. write different official and business letters;
2. write formal and informal emails,
3. Using social media;
4. write a five-paragraph essay;
5. write different reports (visit r., field r., feasibility r., progress r.);
6. write laboratory reports;
7. use the Internet to find specific information;
8. use the computer to write different reports;
9. write minutes of meetings;
10. write a paper on a particular problem or issue;
11. write CVs and applications for work;
12. hold oral presentations;
13. respond to job interviews.

Course content: Introduction to communication skills. Words and sentences. Writing process. Technical information. Internet information and e-mail: World Wide Web (www) and Social media. Essays and exam answers. Laboratory reports. Reports. Proposals, specifications and manuals. Final year projects and reports. Spoken presentations. Group work and meetings. Letters. CVs and job applications. Interviews: Interview for job.

Methods of teaching: 30 hours of lectures + 15 hours of exercises. Approximately 80 hours of personal study and exercise including home-work.

Grading System: Test 1: 25 %,Test 2: 25 %, Test 3: 25%, Homework (seminar paper) 25 %.

Literature:

- Mike Markel & Stuart A. Selber, Technical Communication, 12th Edition, MacMillan, 2018
- John W. Davies, Communication Skills. A Guide for Engineering and Applied Science Students, Prentice Hall, 2011.
- Miller et al, How the World Changed Social Media, UCL Press, 2016
- Majlinda Nishku, Si të shkruajmë: procesi dhe shkrimet funksionale, CDE, Tiranë, 2004.

Course title: German language

Lecturer: Lecturer from UP

Course status: Elective, Semester I, 5 ECTS

Short description: The focus of this course is to teach students to use German to clearly communicate and write engineering technical concepts.

Course goals: Enabling student to actively use the German language to communicate, orally and in writing, at the necessary level for the professional field of electrical and computer engineering.

Expected learning outcomes: Upon completion of this course the student will be able to:

1. actively use German in everyday life.
2. communicate both orally and in writing the German language at a desired level, specifically as it pertains to their professional field.
3. ask and answer questions in the field of electrical engineering in German.
4. translate texts related to the field of electrical and computer engineering.

Teaching methodology: 30 hours of lectures, 15 hours of exercises. Approximately 80 hours of independent work, including a seminar.

Assessment: Seminar 10%, Intermediary Evaluations 30%, Final Examination 70%.

Ratio between theoretical and practical parts of studying: 2:1

Primary literature:

- Leitner, Arnold, *German Made Simple: Learn to speak and understand German quickly and easily*, New York, USA

Course: Practical Mathematics

Lecturer: Prof. Asoc. Dr. Valdete Rexhëbeqaj Hamiti
Course status: Elective, Semester I, 5 ECTS

Short course description: In this course will be studied algebraic expressions, polynomials, equations with an unknown, inequations, arithmetic and geometrical strings, trigonometry and analytic geometry.

Course objectives: Students should be trained so that the knowledge gained through this course can be applied in acquiring knowledge from the following subjects.

Learning outcomes:
Upon completion of this course the student will be able:

- to design and solve different problems in the field of equations with an unknown and their implementation;
- to operate with polynomials;
- to apply basic concepts from analytical geometry and trigonometry to various engineering problems.

Teaching methodology: Lectures, discussions, exercises, consultations, homework, seminar work, intermediate evaluations, final exam.

Evaluation methods: Seminar work and homework (20%), first intermediate evaluation (30%), second intermediate evaluation (30%), final exam (20%).

Concretization tools/IT: Pencil, whiteboard, projector and computer.

Ratio between theoretical and practical part: Ratio between the theoretical and practical part is 1:1.

Literature:

- Hamiti E., Peci H., Loshaj Z., Gjonbalaj Q., Lohaj Sh. - Përmbledhje detyrash nga matematika, Prishtinë 2001.
- M. Berisha, D. Kamberi, R. Gjergji, R. Zejnullahu, Përmbledhje detyrash nga matematika, Prishtinë 1990.

Course title: Calculus II

Lecturer: Prof. Asoc. Dr. Qefsere Gjonbalaj, Prof. Asoc. Dr. Shqipe Lohaj, Prof. Asoc. Dr. Valdete Rexhëbeqaj Hamiti
Course Status: Mandatory, Semester II, 7 ECTS

Course description
In this subject we work: Integral computation and its implementation, functions with two or more variables and ordinary differential equations.

Course objectives
The student should be able to apply the knowledge gained through this course as an auxiliary device in the studies of electrical engineering and computer engineering subjects

Expected results.
After completion of the course, student will be able to

1. Understand the concept of indefinite and definite integral as well as their application in the measurement of various measures in Geometry, Electrotechnics, Telecommunication, Informatics and other fields;
2. Generalize concepts related to functions with one variable into multi variable functions and in particular into those with two variables. Also be able to apply every concept related to the differential calculation for the one variable function in the case of two variable functions;
3. Think logically about various differential equations, solve concrete examples step by step and model different practical problems through differential equations.

Methodology of teaching: Lectures, discussions, exercises, consultations, homework, mid term exams, final exams.

Methods of assessment: Homework and seminar (10%), First periodic exams (25%), Second periodic exams (25%), Final exams (40%).

Concretization tools: pencil, whiteboard, projector and compjuter.

Ration between Theoretical part and exercises: 2:1

References

- Hamiti E. - Matematika I, Prishtinë 2008.
- Hamiti E. - Matematika II, Prishtinë 2008.
- Hamiti E. - Matematika III, Prishtinë 2008.
- Peci H, Doko M. - Përmbledhje detyrash të zgjidhura nga Matematika I, Prishtinë 1997.
- Loshaj Z. - Përmbledhje detyrash të zgjidhura nga Matematika II, Prishtinë 1996.
- Hamiti E, Lohaj SH.- Përmbledhje detyrash të zgjidhura nga Matematika III, Prishtinë 2008.

Course title: Physics II

Lecturer: Dr.sc. Valon Veliu
Course status: Mandatory, Semester II, 6 ECTS

Short course description: The course includes basic knowledge of physics necessary to gain general knowledge that is basic in engineering

Course objectives: Using the physical laws of modern physics in modeling and solving specific engineering problem.

Learning outcomes: On successful completion of the course, students will be able to:

1. Apply the linearization technique to equations of motion of oscillatory.
2. Explain the wave equation in non dispersive medium.
3. Analyze optical systems using the methods of geometrical optics.
4. Explain the phenomena of interference, diffraction and polarization of light.
5. Explain Planck's law of black body radiation.
6. Relate the atomic spectrum to quantization of energy levels..

Teaching methodology: 45 hours of lectures + 15 hours of auditorial exercises + 15 hours of laboratory exercises. Approximately 75 hours of personal study.

Evaluation methods: Seminar 10%, Mid-term exams 30 %, Final Exam 60 %.

Concretization tools/IT: Pencil, whiteboard, projector and computer.

Literature:

- S. Skenderi, R,Maliqi, "Physic for thechnical faculty", UP, Pristina, 2005.
- J. Serway, Physics for scientists and engineering , Thomson Books, 2004.-D. Haliday, R.Resnick, J.Walker, Fundametals of Physics, John Wiley &Sons, 2001

Course title: Fundamentals of electrical engineering 2

Lecturer: Prof. Dr. Luan Ahma, Prof. Dr. Enver Hamiti, Prof. Dr. Mimoza Ibrani, Prof. Ass. Dr. Vjosa Shatri
Course status: Mandatory, Semester II, 7 ECTS

Short Course description: Basics of magnetism. Magnetic flux density. Lorentz force. Biot-Savart's law. Forces due to magnetic field. Magnetic torque and moment. Ampere's law. Magnetic dipole. Magnetization in materials. Generalized Ampere's law. Magnetic boundary conditions. Magnetostatic field analysis using MATLAB. Faraday's law. Inductors and inductances. Magnetic energy. Magnetic circuits. Current and voltage waveforms. Techniques of Circuit Analysis. Sinusoidal Steady-State Analysis. The Sinusoidal Response, The Passive Circuit Elements in the Frequency Domain, Sinusoidal Steady-State Power Calculations, Maximum Power Transfer. AC circuit analysis using SPICE. Magnetically coupled circuits. Response of First-Order *RL* and *RC* Circuits. Transient circuit analysis using SPICE. Balanced Three-Phase Circuits. SPICE analysis of three phase circuits.

Course objectives: The purpose of the course is to introduce the basic principles of magnetic field and AC current circuits analysis.

Learning outcomes: On successful completion of the course, students will be able to:

1. Understand fundamental laws of magnetism.
2. Apply the fundamental laws of magnetism for solving of magnetic field problems.
3. Apply MATLAB software pavckage for solving basic problems in magnetic field.
4. Understand and apply methods for AC circuit analysis such as: Kirchhof's lows, node voltage method, mesh current method, superposition method, Thevenin's and Norton's theorem.
5. Understand transient response of first order circuits (series RL circuits).
6. Apply PSPICE Software for AC circuit analysis.
7. Apply gained knowledge in other electrical engineering fields.

Teaching Methodology: Presentations, tutorials, discussions, and laboratory works meaning**:** 60 hours presentations + 15 hours of tutorials, and 15 hours laboratory works.

Evaluation: First assesment:30%, Second assesment: 25%, Home exercises 10%, Attendance: 5%, Final exam, 30%, Total:100%.

Concretization tools: Computer, video projector, and equiped lab with neccessary devices to ilustrate all teaching material.

Literature:

- Nexhat Orana, Bazat e elektroteknikës 2, Prishtinë, 1994
- M.N. Sadiku, Elements of electromagnetic, Oxford University Press, New York, 2001
- Ch. Alexander, M.N. Sadiku, Electric Circuits, McGraw Hill, New York, 2000.

Course title: Algorithms and Data Structures

Lecturer: Prof. Ass. Dr. Avni Rexhepi, Prof. Asoc. Dr. Kadri Sylejmani
Course status: Mandatory, Semester I, 5 ECTS

The goal: The purpose of the course is to help students in advancing their knowledge for different algorithms, data structures and use of classes and objects in programming.

Learning outcomes: On successful completion of the course, students will be able to compile advanced algorithms, to define structures and classes and to use them in writing advanced programs.

Course content: Definition and use of advanced functions, inline functions, macro functions, function overloading, templates. User defined types. Object oriented programming. Structures: definition of different structures, with different functions as their components, operating structure components. Classes and objects: class definition, classes and member functions. Using public and private members. Declaring objects and operating with their components. Class constructor and destructor. Inheritance, arrays within objects and arrays of objects. Pointers and functions with pointers. References and functions with references. Algorithms, analysis of algorithms, algorithm's growth rate, classification. Data structures. Stack. Queue. Linked lists, adding/deleting nodes. List searching and sorting. Binary tree, tree traversal algorithms, insertion, search and deletion, BST-binary search tree, heap, balanced trees. Graphs, traversal algorithms, minimum spanning tree (Dijkstra-Prim, Kruskal), shortest path algorithm (Dijkstra). Searching and sorting algorithms (Insertion sort, Selection sort, Bubble sort, Shell sort, Merge sort, Quick sort, Heap sort, Bucket sort, Radix sort).

Methods of teaching: 30 hours of lectures + 30 hours of auditorial/lab exercises. Approximately 65 hours of personal study and exercise including home problems/tasks.

Grading System: Attendance lectures 10%, Attendance lab. 10%, Mid-term problems 30 %, Collocui/Final Exam 50 %

Literature:

- Agni Dika, Programimi i Orientuar në Objekte, me programe në C++, UEJL, Fakulteti i Shkencave Bashkëkohore, Tetovë, ISBN 9989-866-25-2, http://www.agnidika.net/programimiobjekte.pdf
- D. S. Malik, C++ Programming: Program Design Including, Data Structures, Course Technology, Thomson Leraning, Boston, Massachusetts, ISBN 0-619-03569-2
- H.M. Deitel, P. J. Deitel, How to Program C++, Prentice Hall, Upper Saddle River, New Jersey, ISBN 0-13-111881-1
- Robert Lafore, Object-Oriented Programming in C++, Sams, Indianopolis, Indiana, ISBN-10:0-672-32308-7
- D. S. Malik, Programming: From Problem Analysis To Program Design, Course Technology, Thomson Leraning, Boston, Massachusetts, ISBN 0-619-06213-4

Course title: Digital Circuits

Lecturer: Prof. Dr. Sabrije Osmanaj, Prof. Ass. Dr. Artan Mazrekaj
Course status: Mandatory, Semester II, 5 ECTS

Short course description: Number systems and codes: number systems, codes, encoding, error detection and correction. **Boolean algebra:** propositional logic, Boolean variables, basic operations, derived operations, axioms and theorems, proofs of theorems. **Boolean functions and logic gates:** representations, methods of simplification and conversion, Karnaugh map and truth table, logic gates and circuits, functionally complete sets of operations, timing hazards, logic families and technologies and their characteristics.
Combinational logic circuits: encoders and decoders, multiplexers and demultiplexers, comparators, adders, multipliers, arithmetic and- logic units. **Computer-aided digital design:** minimizers, schematic editors, circuit simulators, hardware description languages, PCB layout designers, IC layout designers. **Sequential logic circuits:** latches and flip-flops, truth table and excitation table, registers, counters, shift registers, ring counters, excitation equations, state table and state diagram, analysis and synthesis of sequential logic circuits. **Three-state buffers and buses:** buffer, Schmitt trigger buffers, three-state buffers, serial buses, parallel buses. **Programmable logic circuits:** storage matrix, ROM, PROM, EPROM, EEPROM, Flash, PLA, PAL, GAL, SRAM, DRAM, CPLD, FPGA. Use of hardware description languages for implementation of combinational and sequential logic in CPLD and
FPGA circuits **Additional topics (in case of spare time, but not required in exams):** microcontrollers, microprocessors, analog-digital and digital-analog converters, clock generators.

Course objectives: To gain the basic theoretical understanding of functioning of digital structures. To acquire the knowledge and basic experience of practical design, implementation and testing of digital structures.

Learning outcomes: After successful completion of the course, students will be able to:

- describe the basic combinational and sequential structures of digital circuits;
- explain the functioning of such structures;
- describe this functioning in one of the hardware description languages;
- detect flaws in such functioning;
- eliminate the basic among such flaws;
- choose the optimal design of a digital structure, taking into account the requirements of size, cost and reliability of functioning;
- design combinational and sequential circuits using the methods of abstract and structural synthesis;
- build a prototype of the designed structure in the form of a printed circuit board;
- test the functioning of such a prototype;
- Formulate the strengths and weaknesses of the chosen design.

Teaching methodology: Lecture classes with examples of problem solving and tutorials to illustrate the theoretical concepts, laboratory work for acquisition of practical skills in design, implementation and testing of digital structures.

Evaluation methods: Final rating represents the sum of: The successful practical work: 25%, First intermediate evaluation: 15%, Second intermediate evaluation: 20%, Regular attendance and involvement in discussions and seminars 10%, Oral test or final exam: 30%, Total: 100%

Concretization tools/IT: Computer, projector, lab, table.

Ratio between theoretical and practical part: Ratio between the theoretical and practical part is 1:1.

Literature:

- Floyd Thomas L., Digital Fundamentals (10th Edition), Prentice Hall, 2008.
- M. Morris Mano, M. D. Ciletti. Digital Design, 6th ed. Pearson/Prentice Hall, 2017.
- Fundamentals of Digital Circuits, 3rd Edition, by A. ANAND KUMAR, 2014, Delhi.
- S.M. Deokar, A. A. Phadke, "Digital Logic Design and VHDL", Wiles, 2009
- Digital Circuit Analysis and Design with SIMULINK Modeling: And Introduction to CPLDs and FPGAs, Second Edition, Steven T. Karris, Orchard Publications 2007.
- J. F. Wakerly. Digital Design: Principles and Practices, 5th ed. Pearson/Prentice Hall, 2017.
- C. Maxfield. Bebop to the Boolean Boogie, 3rd ed. Newnes, 2009.

Course: Discrete Mathematics and Probability

Lecturer: Prof. Asoc. Dr. Qefsere Doko Gjonbalaj
Course status: Mandatory, Semester III, 5 ECTS

Course status: Mandatory
Course Description: In this course will be studied parts from Mathematical Logic, sets, some important functions, and algebraic structures. Basic concepts of numerical series, polynomial series and functional series. Fourier series. The special attention is paid to the part of the Graph Theory (graphs and trees) and Discrete Probability, where will be studied the basic concepts of probability, discrete and continuous random variables. Numeric parameters of random variables, variance, standard deviation. Then will be explained the discrete distributions, distribution functions and covariance. Basic elements of mathematical statistics, the role of statistics in engineering and statistical tests.

Course objective

Students should be trained so that the knowledge gained through this course can be applied as an auxiliary equipment in electrical and comuter engineering studies.

Learning outcomes:
Upon completion of this course the student will be able to:

1. Understand the role of mathematical logic, finite sets and relations
2. Become familiar with the problems associated with n-ary relations
3. Analyze electrical engineering problems using Furie series
4. Apply Fourier series into concrete professional problems
5. Solve some characteristic problems in graph theory
6. Make mathematical models related to concrete professional problems, using knowledge from graph theory and trees
7. Know the basic concepts of the Probability Theory and Statistics;
8. Know the basic distributions attributes as well as their application to professional problems;
9. Process and present statistical data

Teaching methodology: Lectures, discussions, exercises, consultations, homework, seminar work, intermediate evaluations, final exam.
Evaluation methods: Seminar work and homework (10%), first intermediate evaluation (25%), second intermediate evaluation (25%), final exam (40%).
Concretization tools/IT: Pencil, whiteboard, projector and computer.

Ratio between theoretical and practical part: Ratio between the theoretical and practical part is 2:1.

References

- QefsereDoko Gjonbalaj,*"Matematika III me Matematikë Diskrete (Programi i Kompjuterikë*

s)"; Libër universitar; Universiteti i Prishtinës-FECE; 2017, Prishtinë; ISBN 978-9951-00-132-8; Botimi i dyte

- QefsereDoko Gjonbalaj, Shukri Haxha,"*Permbledhje Detyrash nga Matematika III me Matematikë Diskrete (Programi i Kompjuterikës)"*; Libër universitar; Universiteti i Prishtinës-FECE; 2017, Prishtinë; ISBN 978-9951-00-132-8
- Kenneth H. Rosen; MC GRAW HILL: Discrete Mathematics and its Applications, Fifth Edition 2003, ISBN 0-07-242434-6; USA
- Kenneth H. Rosen; MC GRAW HILL: Student Solutions Guide for Discrete Mathematics and its Applications, Fourth Edition 2003, ISBN 0-07-289906-9; USA
- T.T. Soong "Fundamentals of Probability and Statistics for Engineers" State University of New York at Buffalo, Buffalo, New York, USA, 2004

Course title: Databases

Lecturer: Prof. Dr. Lule Ahmedi
Course status: Mandatory, Semester III, 5 ECTS

The goal: Students will be able to apply the basic concepts and techniques for the design and implementation of database applications.
Learning outcomes: Upon completion of this course, the student shall be able to:

1. Know the techniques of describing and managing a huge set of data based on the relational model.
2. Use the experience gained here to design and implement database projects in the practice.
3. Know the typical problems of databases in the practice, and compare existing systems in order to make the proper solution to overcome those problems.
4. Identify the different relevant aspects during the development of huge projects of databases, and respectively organize the adoption of basic principles.

Course content: A preliminary 21roject topics cover (mainly based on the audience): Introduction and relational model. Relational 21roject21. Datalog: Logical rules. SQL: Simple queries, aggregation, grouping, set operators, embedded queries, database creation and views, population with data. SQL and programming languages. SQL: Integrity and triggers. SQL: Procedures and functions. SQL: Data security. Conceptual design: ER model and transformation into relational model. Formal design: functional dependencies and normal forms. Physical design. Operator / query evaluation. Transaction management. Crash recovery.
Methods of teaching: 30 hours of lectures + 30 hours of laboratory exercises. Approximately 65 hours of personal study and exercise including 21roject work.
Grading System:
Active attendance 10%, Project assignment 3 x 10% = 30%, Final exam 60 %.
Literature:

- Database Management Systems (3rd Edition). Raghu Ramakrishnan, Johannes Gehrke. McGraw Hill, 2002.
- Database Systems: The Complete Book (2nd Edition). Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer Widom. Pearson, 2008.
- Database System Concepts (7th Edition). Avi Silberschatz, Henry F. Korth, S. Sudarshan. McGraw-Hill, 2019.

Additional literature:

- A guide to the SQL standard. C. J. Date, Hugh Darwen. Addison-Wesley Professional, 1996.
- SQL for Web Nerds. Philip Greenspun. http://philip.greenspun.com/sql/

Course title: Object Oriented Programming

Lecturer: Prof. Dr. Isak Shabani
Course status: Mandatory, Semester III, 5 ECTS

Course goal: This course enables students to prepared and successfully apply the concepts and techniques of programming with objects, enabling students to apply object-oriented techniques in software projects. The purpose of the course is to prepare students with modern knowledge in "thinking in object programming", a precondition necessary for complex software systems based on object oriented programming.

Learning outcomes: On successful completion of the course, students will be able to:

1. Understand main concepts of object oriented programming;
2. To write code with classes and use objects;
3. To use inheritance and polymorphism;
4. To be able to predict exceptions and error handling;
5. To be able to create abstract classes and interfaces;
6. Realize e project relating a particular issue using object-oriented programming.

Course content: Introduction to object oriented programming. Introduction to Java. Classes and objects. Java packages. Constructors and destructors. Reference and value types, Data access. Attributes, methods and operators. Delegates and events. Encapsulation, class inheritance, polymorphism. Abstract classes, interface and pattern. Exceptions and error handling. i/o classes, Generic types and methods. Class collection. Testing applications based on object oriented programming. Documenting object oriented programming.

Methods of teaching: 30 hours of lectures + 30 hours of laboratory exercises. Approximately 65 hours of personal study and exercise including homework.

Grading System:
Attendance 10%, Practical project 30 %, Final Exam 60 %

Literature:

- C. Thomas Wu , "An Introduction to Object-Oriented Programming with Java", 2009, Publisher: McGraw-Hill Education; 5th edition
- Daniel Liang, "Introduction to Java Programming", 2014, 10th Edition, Armstrong Atlantic State University
- Danny Poo, "Object-Oriented Programming and Java", 2007, Second Edition 2nd Edition
- Richard Naoufal, "Java: Object-Oriented Programming Concepts", 2018.

Course title: Computer Architecture

Lecturer: Prof. Ass. Dr. Valon Raça
Course status: Mandatory, Semester III, 5 ECTS

The goal: The aim of this course is to introduce students to fundamental concepts of computer architecture and computer hardware/software interface.

Learning outcomes: On successful completion of the course, students are capable of:

1. Describe basic concepts of computer architecture and design.
2. Discuss on current trends of the processor and memory technology
3. Analyze performance issues stemming from different hardware modules of the processors and memory hierarchy
4. Design simple pipelined processors using Verilog for a RISC architecture

Course content: This course presents main concepts of the computer hardware design. It covers four parts:

1. Fundamentals of Computer Architecture includes basic concepts of computer architecture, current trends in processor design, digital logic review, computer arithmetic, designing hardware with HDL, implementing a simple processor via Verilog and testing it in a FPGA.
2. Instruction Set Architecture introduces RISC instruction set architecture and focuses on compiler execution and optimizations. It also includes concepts of single-cycle and multi-cycle datapaths, basic pipelining concepts, data and control hazards.
3. Memory Technology & Interconnect focuses in memory technology and hierarchical design. Caches and cache protocols are discussed in detail. Basic concepts of Interconnect and computer I/O are covered in this part.
4. Superscalars & Multicores includes concepts on superscalar architectures and introduces the basic concepts of instruction level parallelism. It also includes basic principles of multicore design and shared memory model.

Methods of teaching: 30 hours of lectures + 30 hours of classroom and laboratory exercises. Approximately 65 hours of personal study and exercise including projects.

Grading System:
Individual/Group assignments (total: 40%)
- Assignment 1: 20%
- Assignment 2: 20%

Final Exam: 60%

Literature:

- David A. Patterson and John L. Hennessy, Computer Organization and Design: The Hardware/Software Interface, Fifth Edition, 2013 (P&H).
- Robert B. Reese and Mitchell A. Thornton, Introduction to Logic Synthesis using Verilog HDL, 2006 (R&T).

Course title: Electronics

Lecturer: Prof. Asoc. Dr. Qamil Kabashi, Prof. Dr. Milaim Zabeli
Course status: Mandatory, Semester III, 5 ECTS

The goal: To provide an introduction to the basic concepts in the field of electronics. This course will be one of the fundamental courses in all areas of electrical engineering and will prepare students for more advanced electronics courses.

Learning outcomes: On successful completion of the course, students will be able to:

1. understand the basics of electronics within the field of electrical engineering;
2. understand the diode circuits and their models;
3. understand circuits with bipolar and FET transistors and their models;
4. analyze and design transistor circuits for small signals;
5. analyze and utilize operational amplifiers;
6. continue studies in advanced electronics and microelectronics courses.

Course content: Semiconductors, semiconductor diodes, zener diodes, diode circuits, and various diode circuits for signal processing. Bipolar transistors, basic configurations of circuits with bipolar transistors, models for small signals. Basic amplifier configurations: common emitter, shared base, and common collector. Field effect transistor, working principles, small signal patterns. MOSFET transistors. Basic amplifier configurations: with common source, with common gate and with common drain. Operational amplifiers, ideal and realistic features, basic circuits with operational amplifiers, OA applications

Methods of teaching: 30 hours of lectures + 15 hours of classroom and 15 hours of laboratory exercises. Approximately 65 hours of personal study and exercise including projects.

Grading System:
Attendance 10%.
First midterm exam 10%,
Second midterm exam 10 %,
Third midterm exam 10%
Final exam 50%.

Literature:

- Donald Neamen, Electronic Circuit Analysis and Design, McGraw-Hill Education, Second Edition, 2009
- Adel S. Sedra, Kenneth C. Smith, Microelectronic Circuits, Oxford University Press, 2007
- M. Limani, Q. Kabashi Elektronika, Universiteti i Prishtinës, ligjërata të autorizuara, 2018.

Course title: Web programming I

Lecturer: Prof. Ass. Dr. Dhuratë Hyseni
Course status: Mandatory, Semester III, 5 ECTS

The goal: Students will be able to understand basic concepts related to client-server programming paradigms actual on the Web, as well as to provide experience in design and implementation of Web-based systems.
Learning outcomes: Upon completion of this course, the student shall be able to:

1. Demonstrate fundamental knowledge on traditional as well as cutting-edge concepts and technologies on the World Wide Web.
2. Develop Web applications in practice – prior to that, choose the architecture, modeling and techniques that fit to the nature of the application to develop.
3. Distinguish among the advantages and drawbabcks of client versus server paradigms of distributing software components in the Web.

Course content: A preliminary list of topics cover (mainly based on the audience): Introduction to Internet and the WWW. Client-side programming: HTML, CSS, DOM model, and JavaScript. Interfaces: jQuery. Reuse of existing API-s. Added functionality in Web: wikis, blogs/RSS, tagging. Server-side programming: PHP. Databases on the Web: PHP and MySQL. Regular expressions. Cookies. Session control. Re-engineering an application on the Web.
Methods of teaching: 30 hours of lectures + 30 hours of laboratory exercises. Approximately 65 hours of personal study and exercise including project work.
Grading System:
Active attendance 10%, Project assignment 3 x 10% = 30%, Final Exam 60 %.

Literature:

- Fundamentals of Web Development (2nd Edition). Randy Connolly, Ricardo Hoar. Pearson, 2018.
- PHP and MySQL Web Development (5th Edition). Luke Welling, Laura Thompson. Addison-Wesley Professional, 2016.
- Programming the World Wide Web (8th Edition). Robert Sebesta. Addison Wesley, 2014.
- TCP/IP Illustrated, Volume 1: The Protocols (2nd Edition). Chapter 1: Introduction. Kevin R. Fall, W. Richard Stevens. Addison-Wesley Professional Computing Series, 2011.
- Unleashing Web 2.0: From Concepts to Creativity. Gottfried Vossen, Stephan Hagemann. Morgan Kaufmann, 2007.
- A number of resources on the Web.

Course title: Operating systems

Lecturer: Prof. Ass. Dr. Artan Mazrekaj
Course status: Mandatory, Semester IV, 5 ECTS

Course goal: The purpose of this course is to prepared students with modern knowledge of operating systems and enables to apply successfully the concepts of operating systems.

Learning outcomes: On successful completion of the course, students will be able to:

1. To understand main concepts and the structure of operating system;
2. To compare and make distinction among different operating systems;
3. To manage with process conflicts and executing threads;
4. To manage with memory, processor and input/output units;
5. To analyze operating systems and implement a paperwork relating particular issues with operating systems.

Course content: Introduction to operating systems. Operating system structure, Process management: process concepts, threads, process scheduling, process synchronization, deadlocks. Memory management: memory management strategy, virtual memory management. I/O Management: file system, file system implementation, structure of disk data saving into. I/O systems, Operating System Protection. Operating System Security. Analysis and Studies on operating systems: UNIX, Linux, Windows, Minix, Mach, Android..

Methods of teaching: 30 hours of lectures + 30 hours of laboratory exercises. Approximately 65 hours of personal study and exercise including homework.

Grading System:
Attendance 10%, Practical project 30 %, Final Exam 60 %

Literature:

- Abraham Silberschatz, Peter Baer Galvin dhe Greg Gagne, "Operating System Concepts", 10th Edition, 2018.
- Andrew S. Tanenbaum, "Modern Operating Systems", 4rd Edition, 2015.

Course title: Software Engineering

Lecturer: Prof. Dr. Blerim Rexha
Course status: Mandatory, Semester IV, 5 ECTS

The goal: The purpose of the course is to introduce the basic principles of the development of a cost-effective and high-quality software programs.

Learning outcomes: On successful completion of the course, students will be able to:

1. Possess basic knowledge about software processes,
2. Possess basic knowledge on software process models,
3. Apply different software process models,
4. Have basic knowledge of the cost of development of software systems in practice, and
5. Manage software projects.

Course content: Software systems engineering, Ethical and professional responsibility, Organization, people and computer systems. Models of software processes, Iterative and approximate processes, Rational Unified Model, CASE. Management activities, Project planning, Scheduling activities, Risk management, Version planning, Version management, Software tools. Functional and non-functional requirements, User requirements, System requirements, Interface specification, Document of software requirements, Feasibility study, Analysis and validation of the requirements. System model and architectural design. Agile methods of software development. Reusability of software and testing.

Methods of teaching: 30 hours of lectures + 30 hours of lab exercises. Approximately 65 hours of personal study including home/group project assignments.

Grading System: Classroom Assessment 10%, Projects 40%, Final assessment 50 %

Literature:

- Ian Sommerville, Software Engineering, 10th Edition 2016
- Roger S. Pressman, Software Engnineering, A Practitioner's Approach, 8th Edition 2014

Course title: Data Security

Lecturer: Prof. Dr. Blerim Rexha
Course status: Mandatory, Semester IV, 5 ECTS

The goal: To provide students with practical survey of principles and practice of cryptography and data security, smart cards and their practical use.

Learning outcomes: On completion of this course, students should:

1. Have basic knowledge about cryptography,
2. have knowledge about symmetric and asymmetric encryption algorithms,
3. have knowledge about hash algorithms,
4. be able to apply different cryptographic and hash algorithms,
5. have basic knowledge about smartcards and their usage in real life applications,
6. be able to manage public keys, and
7. be able to analyze and understand applications that use cryptographic algorithms.

Course content: Will include: Symmetric encryption algorithm: Data Encryption Standard (DES) an its variants, Asymmetric encryption algorithm: Rivest Shamir Addelman (RSA), Hash functions: MD5, SHA1 and SHA256, Public Key Infrastructure (PKI), Digital signatures, Authentication, Smart cards, Biometric documents, Hands on: using digital certificates for data security.

Methods of teaching: 30 hours of lectures + 30 hours of lab exercises. Approximately 65 hours of personal study including home/group project assignments.

Grading System: Classroom Assessment 10%, Projects 40%, Final assessment 50 %

Literature:

- Bruce Schneier, Applied Cryptography, ISBN 978-1-119-09672-6, March 2015
- Alfred J. Menezes, Paul C. van Oorschot and Scott A. Vanston, Handbook of Applied Cryptography , ISBN: 0-8493-8523-7 1996
- H.X. Mel & Doris Baker, Cryptograpfy Decrypted, 2004
- Mtthew MacDonald & Erik Johansson: C# Data Security, 2003

Course title: Human Computer Interaction

Lecturer: Prof. Dr. Isak Shabani
Course status: Mandatory, Semester IV, 5 ECTS

Course goal: The aim of the course is to përpare students and enable them to understand the human-computer interactions concepts and techniques and equip students with modern knowledge in thinking of human connectivity with the computer through graphical interfaces (GUI) and interaction design.

Learning outcomes: On successful completion of the course, students will be able to:

1. Understand main concepts of human-computer interactions;
2. To know and identify main characteristics of virtual reality;
3. To understand basic concepts on interactions design;
4. To understand human-computer communication paradigms;
5. To analyse, design, implement and evaluate software systems through techniques waterfall model and agile scrum methods.

Course content: Introduction to the human-computer communication subject. Human. Thinking. Computer. Virtual reality. Human-computer interaction. Norman Stages. Paradigms in HCI. Interaction design. HCI in software processes. Realization of graphical user interface (GUI) through the JavaFX framework. Basics of designing interactive systems. The design rules of interactive systems. Support and implementation of interactive systems. Interactive Systems Evaluation Techniques. The universal design of interactive systems. Supporting users in interactive systems.

Methods of teaching: 30 hours of lectures + 30 hours of laboratory exercises. Approximately 65 hours of personal study and exercise including homework.

Grading System:
Attendance 10%, Practical project 30 %, Final Exam 60 %

Literature:

- Alan Dix, Janet Finlay, Gregory D. Abowd and Russell Beale, "Human-Computer Interaction", 2004
- Jenny Preece, Helen Sharp, Yvonne Rogers, "Interaction Design: Beyond Human-Computer Interaction", 2015
- Donald A. Norman, "The Design of Everyday Things", Paperback 2013

Course title: Web programming II

Lecturer: Prof. Ass. Dr. Dhuratë Hyseni
Course status: Mandatory, Semester IV, 5 ECTS

The goal: This course covers advanced Web programming at the client and the server side, advanced and recent models, concepts, and techniques of development in the Web, as well as adaption to and from other platforms on the Web such as mobile devices and social networks.

Learning outcomes: Upon completion of this course, the student shall be able to: 1.

1. Demonstrate knowledge on advanced and latest concepts and techniques on the World Wide Web.
2. Develop advanced Web applications in practice.
3. Put in use advanced approaches of Web applications programming or configuration, such as through frameworks and design patterns.
4. Adapt Web solutions to and from other physical or conceptual platforms, like mobile devices or social networks.

Course content: Object-orientation in the Web. Mediation via XML/JSON. Asynchronous client-server communication: AJAX. Client-side frameworks: Bootstrap. Code re-use using design patterns: MVC. Configuration or programming: CMS-s (WordPress). Server-side frameworks: Laravel. Responsive programming and migration to mobile devices: CSS3 and new mobile behaviour. Advanced multimedia (HTML5 canvas). Web security. Migrating a Web aplication to a social networks. Integrate social networks to a Web application. Latest trends: Node.js and MongoDB.
Methods of teaching: 30 hours of lectures + 30 hours of laboratory exercises. Approximately 65 hours of personal study and exercise including project work.
Grading System:
Active attendance 10%, Project assignment 3 x 10% = 30%, Final Exam 60 %.

Literature:

- Head First Ajax. Rebeca Riordan, O'Reilly Media, 2009.
- Fundamentals of Web Development (2nd Edition). Randy Connolly, Ricardo Hoar. Pearson, 2018.
- Programming the Mobile Web (2nd Edition). Maximiliano Firtman. O'Reilly Media, 2013.
- Programming Social Applications. Jonathan LeBlanc, O'Reilly, 2011.
- A number of resources on the Web.

Course title: Legal, Ethical and Social Issues in ICT

Lecturer: Prof. Ass. Dr. Dhuratë Hyseni
Course status: Elective, Semester IV, 5 ECTS

Course goal: The purpose of the course is to motivate and apply the concept of critical thinking and information technology's impact on personal, societal, legal and ethical issues.
Learning outcomes: On successful completion of the course, students will be able to:

1. Think critically about information technology issues, actively engage others in dialogue about them, and relate them to personal and societal values.
2. Analyzing and critically evaluate ideas, arguments, and points of view.
3. Express a reasoned position on an issue, both orally and in writing.
4. Analyze the relationships among ethical, social, and political issues that are raised by information systems.
5. Identify the main moral dimensions of an information society and specific principles for conduct that can be used to guide ethical decisions.

Course content: Introduction to Ethical Theory; Professional Ethics and Responsibility; Critical Thinking Skills; Technology and Privacy; Technology and Free Speech; Encryption and Communication; Intellectual Property; Cybernetic Crime, Security and Protection; Moral Dimension of Information System; Social Issues and Technology; Professional Ethics.
Methods of teaching: 30 hours of lectures + 30 hours of laboratory exercises. Approximately 65 hours of personal study and exercise including seminars.
Grading System: Seminar 10%, Mid-term exams 30 %, Final Exam 60 %
Literature:

- Sara Baase, A Gift of Fire: Social, Legal, and Ethical Issues for Computing Technology, 4/E, ISBN-10: 0132492679
- Joseph Migga Kizza, Ethical and Social Issues in the Information Age, fourth edition, ISBN 978-1-84996-037-3
- M. David Ermann, Michele S. Shauf; Computers, Ethics, and Society latest Edition, Oxford University Press, ISBN: 0195143027

Course title: Budget and cost analysis

Lecturer: Prof. Ass. Dr. Bahri Prebreza
Course status: Elective, Semester IV, 5 ECTS

Course goal: In this course, students will learn the fundamentals of financial management, cost accounting and budgeting. There by they are able to calculate competitive prices for products and services, budget the projects and manage the finances of an enterprise.
Learning outcomes: Upon successful completion of this module, student will be able to:

1. Know the concept of Finance and Financial Management in a business (Company);
2. Identify undertaken steps for Financial Forecast of a Company;
3. Analyze Financial Statement through financial reports, to make comparisons in time, or between companies;
4. Calculate the break-even point of a manufacturing company;
5. Analyze the budget variance of a cost center.

Course content: Terms used in cost accounting, The Role of Finance & Finance Manager, Financial Forecast, The Time Value of Money, Buying/Selling of Goods and Services, Inventory and Cost of Goods Sold, Break-even point and Budgeting.
Methods of teaching: 30 hours of lectures + 30 hours of laboratory exercises. Approximately 65 hours of personal study and exercise including seminars.
Grading System: Seminar 10%, Mid-term exams 30 %, Final Exam 60 %

Literature:

- Isa Mustafa: Financial Management, RIINVEST, Prishtinë
- Halit Xhafa & Beshir Ciceri: Financial Management, albPAPER-Tiranë,
- Christopher J. Skousen , Larry M. Walther, Managerial and Cost Accounting, 2009

Course title: Microprocessors and microcontrollers

Lecturer: Prof. Ass. Dr. Lavdim Kurtaj
Course status: Mandatory, Semester V, 5 ECTS

Course content: Motivation and introduction to microprocessor systems. Logical circuit and microprocessor system design. Microprocessor systems and details abut communication between units, time diagrams. Internal structure of 8051 and ARM, variations among different producers. Introduction to development system. Instruction set. Input/output ports and communication with peripherals. Programming in assembler and C/C++. Keyboard and display. Program structure. State diagrams. Timer function and programming. Interrupts - generation and processing. Serial communication. A/D and D/A conversion. Programming in higher level languages, Basic, C, C++.

Course objectives: Familiarity with parts of computer hardware. Uses of microprocessors and/ or microcontrollers in different applications. Basic microcontrollers concepts. Familiarity with architecture of 8051 and ARM family of microcontrollers. Familiarity with software development systems. Microcontroller programming in assembler, Basic, C, C++. Programming and utilization of peropheral devices.
Learning outcomes:
On successful completion of the course, students will be able to:

- know structure of microprocessor systems that are faced with;
- design microprocessor and microcontroller based systems for specific application;
- write program for specific application;
- find and repair defects in microprocessor systems.

Teaching methodology: Combined lectures with simulations and demonstrations, discusions, laboratory exams, projects.
Evaluation methods: Intermediary evaluations 15%+15%, Project 40%, Final exam 15%+15%.
Concretization tools: Computer, projector, simulator, experimental development systems.
Ratio between the theoretical and practical part: 40:60

Literature

- S. MacKenzie, The 8051 microcontroller, 4th Edition, Prentice-Hall, 2007
- Muhammad Tahir and Kashif Javed, ARM Microprocessor Systems: Cortex-M Architecture, Programming, and Interfacing, CRC Pess, 2017
- Renesas Synergy Development Kit, User's Manual, Renesas Electronics, 2015
- D. V. Hall, Microprocessors and digital systems, McGraw-Hill
- Muhammed Ali Mazidi, The 8051 Microcontroller And Embedded Systems Using Assembly and C, Pearson Education, 2007
- Vinod G. Shelake, Rajanish K. Kamat, Jivan S. Parab, Gourish M. Naik, Exploring C for Microcontrollers: A Hands on Approach, Springer, 2007
- Manuale të prodhuesëve për mikroprocesor dhe mikrokontroller

Course title: Algorithms Analysis and Design

Lecturer: Prof. Ass. Dr. Avni Rexhepi
Course status: Mandatory, Semester VI, 5 ECTS

The goal: The purpose of the course is to enable students to design algorithms and to do their detailed analysis, regarding efficiency, sustainability, time and space complexity and processor and memory requests. Algorithms for different problems will be defined in C# and/or C++.

Learning outcomes: On successful completion of the course, students will be able to design efficient algorithms based on main parameters for problem solution and to analyse algorithms in detail.

Course content: Algorithm design. Basic analysis of algorithms, input classes, space complexity. Searching and selecting algorithms, sequential search, binary search, selection. Sorting algorithms, design and analysis: insertion sort, bubble sort, selection sort, shell sort, radix sort, heap sort, merge sort, quick sort, polyphase merge sort. Numerical algorithms, calculating polynomials, matrix multiplication, linear equations. Matching algorithms, string matching, Knuth-Morris-Pratt algorithm, Boyer-Moore algorithm, approximate match algorithm. Graph algorithms, graph terminology, graph data structures, depth-first traversal, breadth-first traversal, minimum spanning tree, shortest-path first (Dijkstra's algorithm). Nondeterministic algorithms, NP problems. Other algorithmic techniques, approximation algorithms, TSP Problem, Bin-packing problem, back-pack problem, Graph-Colouring problem.

Methods of teaching: 30 hours of lectures + 30 hours of auditorial/lab exercises. Approximately 65 hours of personal study and exercise including home problems/tasks.

Grading System: Seminar 40%, Final Exam 60 %

Literature:

- Jeffrey J. McConnell, "Analysis of Algorithms, An Active learning approach", Jones and Bartlett Publishers, ISBN: 0-7637-1634-0, 2001.
- Steven Skiena, "The Algorithm Design Manual", Springer Verlag New York Inc., ISBN: 0-387-94860-0.
- D. S. Malik, C++ Programming: Program Design Including, Data Structures, Course Technology, Thomson Leraning, Boston, Massachusetts, ISBN 0-619-03569-2

Course title: Computer Networking

Lecturer: Prof. Dr. Blerim Rexha
Course status: Mandatory, semestër V, 5 ECTS

The aim of the course (module): Introduction to ISO model layers in computer networks, familiarity with communication protocols, description and operation of services at the application, transport, network, and physical layers.

Learning outcomes: After completing this course (module) the student will be able:

1. gain a basic knowledge of protocol layers and services,
2. have basic knowledge of TCP / IP protocols
3. be able to apply protocols,
4. be able to make network configuration,
5. have a basic knowledge of distributed applications in networks,
6. to understand applications based on TCP / IP protocols.

Course content: layer protocols and services, Internet Service Provider (ISP), Internet History. Principles of network applications, Web, HTTP, FTP, email, DNS and web server. Introduction to the transport layer services, multiplexing and de-multiplexing, reliable data transfer, TCP Protocol, wireshark tool for monitoring traffic, routing, network service model, virtual circuits and datagram networks, ports, IP protocol, routing algorithms, routing in the internet, link layer services, error detection codes, CRC, MAC, LAN, Ethernet Hubs & Switches, PPTP protocol, CDMA, WiFi, Mobile IP, cellular architecture, small offices home office network configuration (SOHO)

Methods of teaching: 30 hours for lectures, 30 hours for laboratory exercises. Approximately 65 hours of personal study including home/group project assignments.

Grading System: Classroom Assessment 10%, Projects 40%, Final assessment 50 %

Literature:

- James F. Kurose & Keith W. Ross, "Computer Networking", 7th Ed., Pearson Inc., 2017
- Dougles Comer, "Internetworking with TCP/IP, Principles, Protocols, and Architecture", 2013

Course title: Mobile Device Programming

Lecturer: Prof. Ass. Dr. Dhuratë Hyseni
Course status: Mandatory, semester V, 5 ECTS

The goal: The purpose of the course is to introduce the basic principles of the app development for mobile devices: in Android and iOS. Furthermore, through hand-on lab exercises students will learn about technologies, elements, layouts, and tools to develop native apps for mobile devices.

Learning outcomes: On successful completion of the course, students will be able to:

1. Possess basic knowledge about software architecture for mobile devices,
2. Apply basic techniques for developing apps for mobile devices,
3. Have basic knowledge and apply concepts for Android platform,
4. Posses knowledge for publishing apps in play store,
5. Have basic knowledge and apply concepts for iOS platform, and
6. Have knowledge to develop independently apps for iOS.

Course content: Introduction to mobile devices, Terminology, OO concepts, Introduction in Android, Android architecture, GUI, Layouts, Emulator-Android Virtual Device, Debugging, Android UI – Dialogs, Menus, and WebView, SQL Lite, Adapters and Widgtes, Android Notifications, Threads. Introduction to iOS, Installation and configuration of VM and IDE, Outlets, Actions, and View Controllers, Tab Bar and Navigation Applications , and Application Preferences.

Methods of teaching: 15 hours of lectures + 45 hours of lab exercises. Approximately 65 hours of personal study including two home/group project assignments.

Grading System: Classroom Assessment 10%, two home group assignments 90%

Literature:

- Professional Android X Application Development, Reto Meier, Wrox, 4th Edition, 2018
- Programming Android, Zigurd Mednieks, Laird Dornin, G. Blake Meike, Masumi Nakamura, O'Reilly, 2012
- Objective-C for Absolute Beginners, Gary Bennett, Mitch Fisher, Brad Lees, Apress, 2018
- Beginning Iphone SDK Programming with Objective C, Wei Meng Lee, Wrox, 2010

Course title: Data engineering

Lecturer: Prof. Dr. Lule Ahmedi
Course status: Elective, Semester V, 5 ECTS

The goal: This course covers data engineering in their diversity on models, physical storage, distribution, as well as on their access, integration, or as well manipulation through queries.

Learning outcomes: Upon completion of this course, the student shall be able to:

1. Demonstrate familiarity with advanced data models from object-relational (objects in SQL), semi-structured model (XML), noSQL, distributed in cloud, and in OLAP cubes.
2. Model and manipulate data in whatever of these models through advanced queries in SQL, Xquery, or in noSQL.
3. Develop information systems based in any of these models.
4. Compare and integrate data of these distinct models.
5. Use and get familiar indirectly with new technologies, like Oracle DBMS, programming in Java, and XML and noSQL data manipulation systems.

Course content: A preliminary list of topics cover (mainly based on the audience): 1. Object-relacional databases. Advanced SQL. Optimal physical organisation of databases. 2. XML semi-structured data model: DTD and XML Schema, XQuery. 3. noSQL data model. Queries in noSQL. 4. Distributed (cloud) data. Distributed transaction management. Replication. Distribute queries. 5. Data Warehousing and OLAP. 6. Access to data. Comparison of models. Data integration. Trends and perspectives in the future.
Methods of teaching: 30 hours of lectures + 30 hours of laboratory exercises. Approximately 65 hours of personal study and exercise including project work.
Grading System:
Active attendance 10%, Project assignment 3 x 10% = 30%, Final Exam 60 %.

Literature:

- Principles of Database Management: The Practical Guide to Storing, Managing and Analyzing Big and Small Data (1st Edition). Wilfried Lemahieu, Bart Baesens, Seppe vanden Broucke. Cambridge University Press. 2018.
- Database System Concepts (6th Edition). Abraham Silberschatz, Henry F. Korth, S. Sudarshan. McGraw-Hill Education, 2010.
- Database Concepts (8th Edition). David M. Kroenke, David J. Auer, Scott L. Vandenberg, Robert C. YoderPearson, Pearson, 2018.

Course title: Computer Security

Lecturer: Prof. Dr. Blerim Rexha
Course status: Elective, Semester V, 5 ECTS

The goal: To provide students with advance cryptographic standards and with practical survey of principles and practice of computer security, including viruses, worms, trojans etc. as well as software measures to protect against them.

Learning outcomes: On completion of this course, students should:

1. Have knowledge about cryptography,
2. have knowledge about symmetric and asymmetric advanced encryption algorithms,
3. have knowledge about hash algorithms,
4. be able to apply different cryptographic and hash algorithms to assure computer security,
5. have knowledge about virus, worm and trojan anatomy, and
6. be able to apply various software protection techniques.

Course content: Introduction to cryptography, Terminology, Symmetric encryption algorithm: AES, Asymmetric encryption algorithm: RSA and ECC, Hash functions: MD5, SHA1 and SHA256, Public Key Infrastructure (PKI), Digital signatures, Foundation of Computer Security, Identification and Authentication, Access Control, Unix Security, Windows Security, Database Security, Software Security.
Methods of teaching: 30 hours of lectures + 30 hours of lab exercises. Approximately 65 hours of personal study including home/group project assignments.

Grading System: Classroom Assessment 10%, Projects 40%, Final assessment 50 %

Literature:

- Dieter Gollmann. Computer Security, third edition Wiley, 2011.
- Cryptography and Network Security, by William Stallings, ISBN 10:1-292-15858-1, published by Prentice Hall, 2017.
- https://www.kali.org/kali-linux-documentation/ , 2019
- Matthew MacDonald & Erik Johansson: C# Data Security, 2003

Course title: Software testing

Lecturer: Prof. Asoc. Dr. Kadri Sylejmani
Course status: Elective, Semester V, 5 ECTS

Course goal: This course provides in depth knowledge about techniques for software testing, which aims at preparing students to successfully complete software development projects.

Learning outcomes: On successful completion of the course, students will be able to:
1. Design test cases for various levels of software testing that include unit testing, integration testing, system testing and acceptance testing; 2. Use techniques for black box testing, 3. Use techniques for white box testing, 4. Use various testing tools such as xUnit, NUnit, JUnit, PHPUnit, TestNG, etc.; 5. Stress and overload testing; 6. Perform analysis and static testing.

Course content: Software testing foundations. Software testing cycle. Unit testing. Integration testing. System testing. Acceptance testing. Testing software systems after addition of new modules. Static testing and analysis. Black box testing techniques. White box testing techniques. State transition testing. Stress and overload testing.

Methods of teaching: 30 hours of lectures + 30 hours of laboratory exercises. Approximately 65 hours of personal study and exercise including homework.

Grading System:
Attendance 10%, Practical project 30 %, Final Exam 60 %

Literature:

- Software Testing Foundations. Second Edition , Andreas Spillner, Tilo Linz, and Hans Schaefer. Rocky Nook, Inc. 2007. ISBN 9781 9339 5208 6.
- SOFTWARE TESTING Foundation Guide. Second Edition. Brian Hambling (Editor)
- The Art of Software Testing. Second Edition. Glenford J. Myers, Software Testing and Quality Assurance Theory and Practice. Kshirasagar Naik

Course title: Concurrent Computing

Lecturer: Prof. Ass. Dr. Valon Raça
Course status: Elective, Semester V, 5 ECTS

Course goal: Concurrent Computing course is focused in undertsnading the callenges and problem solving which stem from the implementation of concurrent and distributed programming. This course elaborates the paradigms of concurrent programming including message-passing concurrency, database concurrency and data-parallel concurrency.
Learning outcomes: On succesful completion of this course the student will be able:

- To understand basic concepts of concurrent computing
- To discuss and implement effective ways of structuring concurrent and distributed-memory programs
- To analyze the performance of concurrent programs

Course content: This course presents basic concepts of concurrent programming. It includes four parts:
1. Fundamentals of concurrent computing which includes basic concepts of concurrent computing and focuses on discussing challenges and models of concurrency in programming.
2. Concurrent programming with Erlang presents functional language Erlang for concurrent programming and enables expansion of knowledge in optimization of concurrent programs.
3. Parallelization which focuses on main techniques for parallelizing sequential and concurrent programs.
4. Models, languages and concurrent program verification aims at introducing programming models and languages in the domain of problems which are inherently concurrent and parallel.

Methods of teaching: 30 hours of lectures + 30 hours of laboratory exercises. Approximately 75 hours of personal study and exercise including seminars.
Grading System: Projects 40 %, Final Exam 60 %

Literature:

- M. Ben-Ari: Principles of Concurrent and Distributed Programming (2nd Edition), 2006
- Herlihy, Shavit: The Art of Multiprocessor Programming (1st Edition), 2008
- F. Hébert: Learn You Some Erlang for Great Good! A Beginner's Guide, No Starch Press, 1st edition, 2013

Course title: Entrepreneurship and innovation

Lecturer: External Lecturer
Course status: Elective, sem. V, 5 ECTS

Course goal: The course is taught through a real-life customer development context where students acquire the skills and know-how to develop their business idea all the way from the conceptual stage to the market place up to their skills for entrepreneurship and innovation. This course articulates a space of innovation that aims to be the seed of incubator projects of entrepreneurship in Kosovo. The aim of this course is that students are able to realize their dreams and develop the skills to achieve them in Innovation and Entrepreneurship . The course is designed to encourage students' abilities to think differently to developing innovative products, services and organizations.
Learning outcomes: Upon successful completion of this module, student will be able to:

1. Working in interdisciplinary, in real and virtual environs;
2. learning abroad in a multicultural environment;
3. knowledge transfer in daily mentoring by top experts;
4. gamifying the learning experience;
5. learning together with entrepreneurs;
6. imposed creativity for excellent time and resources management;
7. creating and exploiting an invaluable network of like-minded ambitious talents.

Course content: Promote a culture of creativity, innovation and collaboration. Promoting entrepreneurial culture both professionally and student. Generating, disseminating and sharing knowledge about innovation and entrepreneurship. Develop strategic alliances with agents that structure the ecosystem of innovation and entrepreneurship. Building transnational networks with emblematic actors. Recognize tools and strategies to apply to start-up and projects. Apply new concepts and theoretical frameworks own strengths and knowledge. Managing creativity and innovation. Failure and learn from failure. Understanding and managing innovation processes and their impacts to the environment.
Methods of teaching: 30 hours of lectures + 30 hours of laboratory exercises. Approximately 65 hours of personal study and exercise including seminars.
Grading System: Seminar 10%, Mid-term exams 30 %, Final Exam 60 %

Literature:

- John Bessant, "Entrepreneurship and innovation", 2019, Publisher: John Wiley & Scons Inc.
- Peter F. Drucker, "Innovation and Entrepreneurship, 2006
- Charles Hampden-Turner, "Teaching Innovation and Entrepreneurship", 2009, Cambridge University Press
- Lubar and Halpern, "Leadership Presence", 2004

Course title: IT project management

Lecturer: Prof. Ass. Dr. Dhuratë Hyseni
Course status: Elective, Semestri V, 5 ECTS

Course goal: In this course, students will learn the fundamentals of financial management, cost accounting and budgeting. There by they are able to calculate competitive prices for products and services, budget the projects and manage the finances of an enterprise.

Learning outcomes: Upon successful completion of this module, student will be able to:

1. Understand and are able to explain project management processes;
2. Understand and are able to explain different Agile methodologies (SCRUM, Kanban, Extreme Programming, Lean). Knows advantages and disadvantages of these methodologies.;
3. Have deep understanding of the Scrum project management method. Correctly use project management terminology.
4. Choose appropriate project management methods depending on the project;
5. Plan project activities regarding time and budget;
6. Apply the SCRUM methodology in practice and use Planio.io tool;
7. Use MS Project package for classical project management processes;
8. Smoothly and suggestively express project and its idea;
9. Prepare project documentation;
10. Communicate with each other through group work;
11. Work in a team, assume responsibility for the quality of assigned task;
12. Manage project team, responsibly assess the team leader, the members;
13. Understand the importance of development skills for their professional growth,
14. Develop the need to independently improve their project management skills.

Course content: Introduction to project management. Project integration and scope management. Project time management. Project cost management. Project quality management. Project Human Resources and Communication Management. Project risk management. Agile project management. Different project management methodologies. Components of Scrum. Process: sprints. Reporting. Comparison of different Agile methodologies.
Methods of teaching: 30 hours of lectures + 30 hours of numerical and laboratory exercises. Approximately 65 hours of personal study and exercise including seminars.
Grading System: Seminar 10%, Mid-term exams 30 %, Final Exam 60 %
Literature:

- Schwalbe, K. (2015). Information Technology Project Management. 8th edition. Cengage Learning
- Robert, K. Wysocki (2013). Effective Project Management: Traditional, Agile, Extreme. 7th Edition. Wiley.
- Kenneth, S., R. (2012). Essential Scrum: A Practical Guide to the Most Popular Agile Process. Addison-Wesley.

Course title: Distributed systems

Lecturer: Prof. Dr. Isak Shabani
Course status: Mandatory, Semester VI, 5 ECTS

Course goal: The aim of this the course is to enable and prepared students to understand and apply distributed system techniques.

Learning outcomes: On successful completion of the course, students will be able to:

1. To understand main concepts and system model of distributed;
2. To compare distributed system;
3. To creates distributed systems;
4. Manage distributed objects;
5. To realize e project relating a particular issue with distributed systems.

Course content: Characterization of distributed systems. System models. Networking and internetworking in Distributed Systems. Interprocess Communications. Remote Invocation. Indirect Communication. Distributed components and objects. Web Services. Peer-to-peer systems. Distributed File System. Name Services. Global Time and State. Distributed Transactions. Distributed Replication. Web based distributed systems.

Methods of teaching: 30 hours of lectures + 30 hours of laboratory exercises. Approximately 65 hours of personal study and exercise including homework.

Grading System:
Attendance 10%, Practical project 30 %, Final Exam 60 %

Literature:

- George Coulouris, Jean Dollimore, Tim Kindberg, Peter Baer Galvin dhe Greg Gagne, "Distributed Systems – Concepts and Design", 5th Edition, 2012.
- Andrew S. Tanenbaum, "Distributed Systems – Concepts and Paradigms", 2rd Edition, 2007.
- Manish Varshnev and Shanoo Agarwal, "Concepts of distributed system", 2016

Course title: Data mining

Lecturer: Prof. Dr. Lule Ahmedi
Course status: Elective, Semester VI, 5 ECTS

The goal: Data mining is a relatively young but rapidly growing field that is concerned with developing techniques to among others assist businesses to make intelligent use of their repositories. For example a supermarket might gather data on customer purchasing habits. Using data mining techniques, the supermarket can determine which products are frequently bought together and use this information when planning their business. This course will examine methods on recognizing patterns and relationships among available data, and making predictions from an applications perspective. Experimentation with algorithms for data mining using a programming language will also be provided. Some often problems occuring during data mining will also be addressed.

Learning outcomes: Upon completion of this course, the student shall be able to:

1. Gain solid background on traditional approaches of data mining starting from data processing, to description of existing data or prediction of unseen data using the algorithms in use of data mining.
2. Be capable of developing a basic data mining system.
3. Be able to first choose the right approach of data mining, and then address some common problems.

Course content: Topics include but are not restricted to: data preparation for mining, classification, association rules, clustering, overfitting, the imbalanced classes problem, anomaly detection.

Methods of teaching: 30 hours of lectures + 30 hours of laboratory exercises. Approximately 65 hours of personal study and exercise including project work.

Grading System:
Active attendance 10%, Project assignment 3 x 10% = 30%, Final Exam 60 %.

Literature:

- Introduction to Data Mining (2nd Edition). Pang-Ning Tan, Michael Steinbach, Anuj Karpatne, Vipin Kumar. Pearson, 2018.
- Python Data Science Handbook: Essential Tools for Working with Data (1st Edition). Jake VanderPlas. O'Reilly Media, 2016.

Course title: Internet Security

Lecturer: Prof.Dr. Blerim Rexha
Course status: Elective, sem. VI, 5 ECTS

The goal: Introducing the forms of attacks, algorithms for encryption / decryption, protocols for sending data in secure way through network, Firewalls, Viruses / Trojans, Wireless Security, IPsec.

Learning outcomes: On successful completion of the course, students have knowledge about:

1. forms of attacks, theft of passwords, Social Engineering, Authentication Failures, Protocol Failures, Active and Passive Attackers;
2. Symmetric / Asymmetric algorithm and to make the analysis of the safety of these algorithms;
3. protocols for sending data in the safe mode, using of digital certificates, Internet Mail Architecture;
4. Characteristics of firewall, Types of firewall, Firewall Location and Configuration, Proxy servers;
5. Security in Wireless LAN, Architecture models, Operation Phases, WAP architecture, Cryptographic Algorithms;
6. KALI Linux tolls and explore top 10 vulnerabilities defined OWASP.

Course content: Introduction to Cryptography, Terminology, types of attacks. Symmetric algorithms: AES. Non-symmetric algorithms: Principles of Cryptosystems with Public Key, RSA, Diffie-Hellman, Elliptic Curve, Hash functions (one way functions): MD5, SHA-1, SHA-256, Digital Signatures: Security in the Transport layer, Functionality of MIME and S / MIME, Architecture of Internet Mail, DKIM Strategy, E-mail Threats. Firewall Characteristics, Needs for firewall, Firewall configuration, Creation of a firewall to filter through a Linux PC, De-militarized Zone (DMZ), VPN. Introducing Network Components and Architecture Model, Services, Operation phases, Authentication phases, Key Management phases, Data transfer protection phases, WAP architecture, WAP End-to-End Security. Introduction to IP Security and benefits, Using KALI Linux, OWASP vulnerabilities.

Methods of teaching: 15 hours of lectures + 45 hours of lab exercises. Approximately 65 hours of personal study including two home/group project assignments.

Grading System: Classroom Assessment 10%, two home group assignments 90%

Literature:

- Cryptography and Network Security, by William Stallings, ISBN 10:1-292-15858-1, published by Prentice Hall, 2017.
- Internet Security, by Man Young Rhee, ISBN=0-470-85285-2, published by John Willey & Sons, 2003.
- Penetration Testing: A Hands-On Introduction to Hacking Georgia Weidman, 2014
- https://www.kali.org/kali-linux-documentation/ , 2019

Course title: Big Data

Lecturer: Prof.Dr. Lule Ahmedi
Course status: Elective, Semester VI, 5 ECTS

The goal: The main objective of this course is for the students to get familiar with main models and techniques of storing and manipulating Big Data.
Learning outcomes: Upon completion of this course, the student shall be able to:

1. Gain solid knowledge in gathering and integrating Big Data.
2. Be able to store and manipulate through queries data of distinct models noSQL, and stream data.
3. Get familiar with storage and manipulation systems of Big Data (Apache Spark).
4. Get acquainted with Data Science phases, as well as economic and ethical aspects of working with Big Data.

Course content: Topics include: 1. Sources of Big Data: Machine-generated, organization-generated, generated by people. Integrating diverse data. The 5 v's of Big Data. 2. NoSQL storage solutions (Cassandra) with critical features: speed of reads and writes, and ability to scale to extreme volumes. Memory resident databases (VoltDB, SciDB) and graph databases (Ne4J). Queries. 3. Data streams: Sampling methods. Queries. 4. Systems: Hadoop, HDFS. MapReduce. Cloud computing services. SQL on Hadoop (HBase, Pig, Hive). Apache Spark (Spark Core, Spark SQL, Spark Streaming and GraphX). 5. Data science process. Economic perspective (ROI) on analytics. Privacy and security.
Methods of teaching: 30 hours of lectures + 30 hours of laboratory exercises. Approximately 65 hours of personal study and exercise including project work.
Grading System:
Active attendance 10%, Project assignment 3 x 10% = 30%, Final Exam 60 %.

Literature:

- Principles of Database Management: The Practical Guide to Storing, Managing and Analyzing Big and Small Data (1st Edition). Wilfried Lemahieu, Bart Baesens, Seppe vanden Broucke. Cambridge University Press. 2018.
- Spark: The Definitive Guide: Big Data Processing Made Simple (1st Edition). Bill Chambers, Matei Zaharia. O'Reilly Media, 2018.

Course title: Information Retrieval

Lecturer: Prof. Dr. Lule Ahmedi
Course status: Elective, Semester VI, 5 ECTS

The goal: This course is an introduction to the traditional text Information Retrieval (IR) and the basics of Web IR.
Learning outcomes: At the end of this course, students will:

1. Gain solid knowledge in traditional information retrieval from text.
2. Be capable to design, implement, and evaluate IR systems in the Web, like search engines.
3. Understand the theories behind modern Web search engines, like Google, Swoogle, etc.

Course content: Topics include: Boolean retrieval, vector space model, as well as tolerant retrieval; The Web size estimation and duplicate detection; Link analysis and crawling; Applications of classification and clustering in the IR domain. Additional possible topics: Text-centric XML indexing and ranked retrieval, user interfaces for IR. The course will also be accompanied with demonstration of how these IR concepts and techniques are implemented in modern search engines like Google, AltaVista, Bing, and Clusty.
Methods of teaching: 30 hours of lectures + 30 hours of laboratory exercises. Approximately 65 hours of personal study and exercise including project work.
Grading System: Active attendance 10%, Project assignment 30%, Final Exam 60 %.

Literature:

- 1. Introduction to Information Retrieval. Christopher Manning, Prabhakar Raghavan and Hinrich Schtze, Cambridge University Press. 2008; on-line at http://www-csli.stanford.edu/~schuetze/information-retrieval-book.html
- 2. Information Retrieval (online book), by C. J. van Rijsbergen, (available online here): http://www.dcs.gla.ac.uk/Keith/Preface.html
- Search Engines: Information Retrieval in Practice by W. B. Croft, D. Metzler, and T. Strohman, Pearson Education, 2015.

Course title: Visual Computing

Lecturer: Prof. Ass. Dr.Artan Mazrekaj
Course status: Elective, Semester VI, 5 ECTS

The aim of the course (module): The aim of the course is to prepare the student for independent work in computer graphics, data visualization and image processing.
Learning outcomes: After completing this course the student should be able to:

1. Recognize the graphics hardware devices.
2. To be able using programming languages (eg. C #, Java) to write different codes for graphics processing.
3. Be able to analyze programs and modify them.
4. To be able to use Open Source software as Open GL for graphics data processing.
5. To be able to create 2D and 3D applications and create animations.
6. To be able for image processing.

Course content: Software application for CAD and technical documentation. Hardware units for graphical presentations. Raster-Scan system. The color system. Halftoning approximation. Colors diagram: CIE, RGB, YIQ. Programming primitive objects. Objects: GRAPHICS, POINT, RECTANGLE, COLOR, FONT, PEN, GRAPHICSPATH, BRUSH, SOLIDBRUSH, etc. Drawing different shapes. Permanent drawing. Drawing methods: DRAWLINE, DRAWRECTANGLE, DRAWELLIPSE, DRAWPIE, DRAWPOLYGON, DRAWCURVE, DRAWBEZIER, DRAWSTRING, etc. Algorithms for drawing 2D objects. DDA algorithm. Algorithm middle point of the line. Algorithm midle point of the circle. Two dimensional geometric transformations. Moving, Scaling, Rotation. Homogeneous coordinates for matrix presentation of transformations. 3D graphics. Animations. Tasks and problems solved in C # and OpenGL. Data visualization. Image processing (foundamentals of Matlab).
Methodology of teaching:
30 hours of lectures, 30 hours of laboratory exercises. Approximately 65 hours of independent work including independent projects.
Grading System:
Attendance 10%, Projects 30% , Final Exam 60%

Literature:

- Donald Hearn, M. Pauline Baker, "Computer Graphics - C Version", Prentice-Hall International, 1997
- Peter Shirley, "Fundamentals of computer graphics", 3rd edition, 2009
- T. Theoharis, G. Papaioanou, N. Square there, N. Patrikalakis, "Graphics & visualization - principles & Agorithms" A v. Peters, Ltd.. In 2008.

Course title: Cloud Computing

Lecturer: Prof. Ass. Dr. Artan Mazrekaj
Course status: Elective, Semester VI, 5 ECTS

The goal: The purpose of the course is to equip students with basic knowledge of virtualization and cloud integration services. Knowing and understanding the platforms and communications in the cloud environment. Efficient use of hardware resources, assessment of the performance of the Cloud resources.

Learning outcomes: On successful completion of the course, students will be able to:

1. To understand the key concepts and structure of the cloud.
2. To compare and analyze the methods and algorithms for cloud exploitation.
3. To configure and build a cloud platform by familiarizing in depth with service-oriented architecture.
4. To understand the role and capabilities of cloud services integration.
5. To configure platforms in order to have an efficient use of resources.

Course content: Introduction to Cloud Computing. Types of Cloud Computing. Cloud Computing Models. Virtualization, Infrastructure as a Service- IaaS. Platform as Service - PaaS. Software as Service - SaaS. Services Oriented Architecture (SOA). Migration of Cloud Services. Managing the SLA. Cloud Resource Allocation. Introduction to big data management. Cloud Security. Cloud Challenges. Case studies with different frameworks. Analysis and study on cloud computing platforms, using of simulators, etc.

Methods of teaching: Lectures, laboratory exercises/simulations. Personal study by students and independent project work.
Grading System: Presence 10%, Projects 40 %, Final Exam 50 %

Literature:

- Sandeep Bhowmik, "Cloud Computing", Cambridge University Press, 2017
- Edited by: Rajkumar Buyya, James Broberg, Andrzej Goscinski, "Cloud Computing:Principles and Paradigms", Wiley, 2011.

Course title: Parallel Computing
Lecturer: Prof. Ass. Dr. Valon Raça
Course status: Elective, Semester. VI, 5 ECTS

The goal: The aim of this course is to give a general understanding of the basic concepts of Parallel Computing..
Learning outcomes: On successful completion of the course, students are capable of:

1. Discussing the basic concepts of Parallel Computing including: parallelism, concurrency, dependencies, communication, coordination and synchronization
2. Explaining major parallelization strategies of software parallelization and get familiar with the concept of performance analysis in software programs.
3. Programming applications using parallel methods and optimizing computer programs

Course content:: This course presents main concepts of parallel computing. It consists of three parts:

1. Parallelism fundamentals includes basic concepts, concurrent programming and multithreading in C++, loop parallelization, data sharing, synchronization and parallel task programming.
2. Programming models focuses in shared memory programming and distributed memory programming.
3. Parallel programming and algorithms includes parallel programming in Java, basic concepts of the theory of parallel algorithms and selected topics in Parallel Computing

Methods of teaching: 30 hours of lectures and 30 hours of laboratory exercises. 65 hours of independent work including assignments and projects.
Grading System: Attendance mandatory in lectures, laboratory, tests and assignments.
Individual/Group projects (total: 60%)

- Project 1: 20%
- Project 2: 20%
- Project 3: 20%
- Final Exam: 40%

Literature:

- Introduction to High Performance Scientific Computing - Victor Eijkhout, 2015
- C++ Concurrency in Action – Anthony Williams, 2019
- Using OpenMP - Barbara M. Chapman, MIT Press, 2007
- Using MPI Portable Parallel Programming with the Message-Passing Interface – William Gropp et.al., 1999.

Course title: Data Communication

Lecturer: Prof. Asoc. Dr. Bujar Krasniqi
Course status: Elective, Semester. VI, 5 ECTS

The goal: The purpose of the course is to teach students to understand and apply data communication.
Learning outcomes: On successful completion of the course, students will be able to:

1. To describe data networks and apply data communication over them.
2. Be able to understand the basic concepts and terminology of data transmission.
3. To classify transmission media and common use possibilities.
4. Become familiar with architecture and functionality of fixed networks (Optical, HFC-Hibrid Optical Coax) and mobile (2G, 3G, 4G and 5G) for data communication.
5. To recognize wireless sensor networks, personal based on Bluetooth and WLAN.
6. Explain QoS (Quality of Service) and dependency of services on QoS parameters.
7. To be able to describe and apply IoT (Internet of Things) technology for data communication.

Course content: Data communication, data network and internet. Concepts and terminology of data transmission. Transmission media. Multiplexing techniques. Fixed optical and hybrid optical networks. Mobile networks (2G, 3G, 4G and 5G). Wireless sensor networks. Personal area networks based on Bluetooth technology and WLAN. Quality of Service (QoS). Services offered by fixed and mobile networks. Internet of Things (IoT). Applying IoT for data communication.
Methods of teaching: 30 hours of lectures + 30 hours of laboratory exercises. Approximately 65 hours of personal study and exercise including projects.
Grading System: Presence 10%, Projects 40 %, Final Exam 50 %
Literature:

- William Stallings "Data and Computer Communications", Tenth edition Pearson 2014
- S. Cirani, G. Ferrari, M. Picone, L. Velteri "Internet of Things, Architecturesm, Protocols and Standards", Wiley, 2019
- S. Vijayalakshmi, S. Muruganand ''Wireless Sensor Networks, Architecture, Applications, Advancements'' Mercury Learning and Information 2018.

Course title: Biomedical Engineering

Lecturer: From industry
Course status: Elective, sem. VI, 5 ECTS

The goal: Introducing to biomedical engineering as field, areas of focus and practical applications, using bio-sensing and applying machine learning to biomedical engineering problems.

Learning outcomes: On successful completion of the course, students have knowledge about:

1. Biomedical engineering as a field, areas of focus in research and business;
2. Practical applications of biomedical engineering;
3. Sensors and instrumentation, types of biosensors used for collecting data in the domain;
4. Collecting, annotating and processing data from biosensors and images, building a pipeline of machine learning models to infer and predict

Course content: Introduction to problems in biomedical engineering, terminology, engineering approaches to problems in biomedical engineering, application of biomedical engineering, rehabilitation, performance enhancement, remote monitoring, bio-sensing, EEG (electroencephalography), EMG (electromyography), GSR (electrodermal activity), eye tracking (pupil, saccades, fixations), heart rate monitoring, heart rate variability, respiration, biomedical imaging, brain imaging, cardiac imaging, data collection, experiment setup, data annotation, data processing, window-based approach, feature extraction, time-domain features, frequency domain features, using machine learning algorithms (support vector machine, logistic regression, random forest, xboost, k-nearest neighbors, convolution neural networks, long short-term memory architectures) to build prediction/classification models.

Methods of teaching: 30 hours of lectures + 30 hours of lab exercises. Approximately 65 hours of personal study including two home/group project assignments.

Grading System: Classroom Assessment 10%, Projects 40%, Final assessment 50 %

Literature:

- Enderle, John, and Joseph Bronzino. Introduction to biomedical engineering. Academic press, 2012.
- Wilson, Jon S. Sensor technology handbook. Elsevier, 2004
- Bishop, Christopher M. Pattern recognition and machine learning. Springer, 2006.
- Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. Deep learning. MIT press, 2016.

Course title: Biometrics & Forensics

Lecturer: Prof.dr. Blerim Rexha
Course status: Elective, Semester VI, 5 ECTS

The goal: To provide students with actual biometric technologies utilized by personal documents, actual standards and regulations as well as cybernetic crimes and their detection and protection from them.

Learning outcomes: On completion of this course, students will be able to:

1. Apply biometric technologies for digital identity and modern symmetric encryption algorithms,
2. Apply and understand the latest legal and technical standards on identification documents,
3. Apply public key algorithms, use and understand digital signatures and public key infrastructure,
4. Have knowledge about latest cyber-attacks,
5. Use different forensic tools, and
6. Collect and document digital proves

Course content: Will include: Concepts and trends in document identity, Biometric properties, legal and technical standards, ICAO and IEC, Secure communication protocols, EU experience in eID security, Computer forensics, Preparing the environment, Evidence gathering, Data extraction, Encryption and password policy, Forensic tools and final reports.

Methods of teaching: 30 hours of lectures + 30 hours of lab exercises. Approximately 65 hours of personal study including home/group project assignments.

Grading System: Classroom Assessment 10%, Projects 40%, Final assessment 50 %

Literature:
Essential:

- Walter Fumy, Manfred Paeschke, Handbook of eID Security , ISBN=078-3-89578-379-1, 2011
- Digital Forensics with Open Source Tools. Cory Altheide and Harlan Carvey, ISBN: 978-1-59749-586-8, Elsevier publication, April 2011
- Computer Forensics and Cyber Crime: An Introduction (3rd Edition) by Marjie T. Britz, 2013.

Course title: Internship

Lecturer: Prof. Ass. Dr. Sevdie Alshiqi

Mandatory, Semester VI, 5 ECTS

Course goal: The students gain professional experience in the field of computer and software engineering in one of the local or international companies/institutions under supervision from an expert from the field.

Learning outcomes: To be qualified for professional work in the field of computer and software engineering, hence become better prepared for the labor market.

Course Content: The content of this course depends on the company where the student shall finish 120 working hours. As a result, the content is drafted jointly from the coordinator of professional internships, appointed from the company, and the student who is going to work in such company. The coordinator of professional internships, who is appointed from the company, guides the student throughout the duration of his/her work in the company, and, at the end, he confirms the successful completion of the students' internship by signing the exam grade sheet.

Methods of teaching: 120 hours work in a company and for writing the seminar.

Grading System: Seminar Work 40%, Defense 60%. TOTAL: 100%.

Course title: Diploma Thesis

Course status: Mandatory, Semester VI, 5 ECTS

Course goal: Diploma thesis is a comprehensive and independent assignment, where student has to demonstrate the capability to analyze the problem from theoretical or practical aspect., in order ot provide a solution by applying acquired knowledge from differen courses and literature.

Learning outcomes: On successful completion of this module student will be able:

- To gain confidence on acquired knowledge
- To get skills for further study from literature
- To consult with the supervisor
- To present their work in a written form, and standard language
- To present their findings in an oral evaluation

Course Content: Diploma thesis topic may be proposed by the supervisor, or be chosen by the student. It has to be compatible with the qualification profile of the student.

Methods of teaching: Determined by the regulation for diploma thesis of the favulty.

Literature:

- Depending on the topic of the diploma thesis, literature will be provided by the supervisor.