EUROPEAN UNIVERSITY OF LEFKE

DEPARTMENT OF SOFTWARE ENGINEERING

PROGRAMME HANDBOOK

2023

TABLE OF CONTENTS

Department of Software Engineering

The Department of Software Engineering was founded in 2008 and continues its education as one of the five engineering programmes currently offered in the Engineering Faculty at EUL. The programme accepted its first students in the fall semester of the 2008-2009 academic year, and the first group of graduates graduated in the spring semester of the 2012- 2013 academic year. The medium of instruction is English. The university awards a 4-year Bachelor-of-Science (BSc) degree in Software Engineering upon completion of the programme's curriculum that corresponds to 142 EUL credits (240 ECTS). In the current credit system, for each course the evaluation criteria for the semester (midterm exams, homework, laboratory, presentation, project and course participation, final exam, make-up exam, re-sit exam) and their weights are determined by the lecturer and acknowledged by the department.

The department is recognized by the Higher Education Council of Turkey (YÖK) Ref Number: B.30.0. EÖB.000.00.03-01.01-2224. The department offers formal education in English. The Software Engineering department has been accredited by ASIIN since 2017.

The diplomas explicitly state the length of the education (4 years) and the degree awarded. EUL Software Engineering Department's academic year includes two semesters, Fall and Spring, each lasting at least 14 weeks. The academic calendar to be used in conjunction with the program is determined every year with the University Senate decision. Some classes may be reoffered in the so-called "Summer School". The purpose of the Summer School is to offer classes for students who has taken and failed, who hasn't taken, or who are having to retake some courses to raise CGPA. Summer School has a compressed 7-week duration, where the weekly class hours have been doubled. The Software Engineering Department program is a program supported by practical, laboratory and internship studies, as well as theoretical teaching. In this way, students are given the ability to use the theoretical knowledge they have acquired in solving real-life problems. The Software Engineering Department uses the already set-up laboratories together with the sister departments (Computer Engineering Department & Management Information Systems Department). The Software Engineering Department has developed its mission and vision statements in line with those established by the University and published them on the University's web- site.

Mission

The mission of the Software Engineering Programme is to produce graduates that are creative and who have the mathematical, analytical, programming, communication, critical thinking, leadership and decision-making skills necessary in leading a successful career, whilst being aware of the social and ethical aspects of their work.

Vision

The vision of the Software Engineering Programme is to become the prominent and preferred engineering programme in the Eastern Mediterranean region, producing highly competent graduate software engineers who are able to follow the latest developments in the field.

We see our graduates as qualified and in high demand as professional software engineers with a solid scientific background and ethical and moral values.

Aim of the Programme

The main aim of the programme is to provide our students with a well-anchored, solid foundation in the software engineering profession. Software engineering involves the design, production and operation of modern software systems based on engineering principles. Our programme emphasises the requirements demanded in the software engineering field, which is fulfilled via the understanding of the software lifecycle, the analysis, specification, design, implementation and management of software systems. The programme blends engineering, programming, project leadership and software construction. Focusing on the importance of working as part of a team and able to recognise projects that demand teamwork, our programme is tailored to give a well-balanced education, one that strikes a good compromise between practical and theoretical concepts. In the sequel, an intended qualifications profile is established by defining the programme educational objectives and learning outcomes. The competence profile has followed on from the Computer Engineering Department (CE), which has previously been accredited by MÜDEK (Association for Evaluation and Accreditation of Engineering programmes) and also holds EUR-ACE label, with emphasis on the topics in the information technology sector. Besides to this, Software Engineering Department has been accredited from ASIIN on 2017 and also was awarded EURO-INF label.

The improvement of the program follows the feedback and includes the topics where there is major on-going research. In addition, the program outputs were determined and decided in the meetings attended by all lecturers in order to meet the opinions and expectations of external stakeholders and to reveal the difference from the Software Engineering programs of other universities and institutions. Program outputs are reviewed and updated if necessary, taking into account academic student meetings, graduate student correspondence, sector/consultant surveys, alumni surveys and the opinions of all stakeholders in a 4-year period. Moreover, the review of educational objectives is also occurred with similar time periods. Graduates of the Software Engineering Programme are equipped with the knowledge and skills demanded by the sector. Analyzing, designing, implementing, testing, deploying, managing and maintaining software requires the knowledge and skills of a software engineer. Given that the world today depends so much on large-scale complex software systems, it is not a surprise that qualified software engineers are in big demand, and well compensated financially.

Software engineers may find employment as the system or business analyst, quality assurance and test engineer, web or mobile applications developer, front-end or back-end developer, systems or

applications software developer, database administrator, and Information Technology (IT) specialist or consultant. Employers of software engineers include banking and finance sector, industry, large international companies with IT departments, the public or private sector and techno-parks at universities. Graduates of our programme may also find jobs in different fields such as the education and media. Our graduates can also pursue postgraduate studies in a relevant area such as the Computer Engineering, Computer Science, Management Information Systems and Information Technology.

According to the vision and mission of the department, an intended qualifications profile has been established which has been summarized in educational objectives and defined in thirteen program outcomes. In accordance with the mission and vision statements, the alumni (graduate) and sector surveys the educational objectives of the programme are established as follows:

- EO1: To be practicing software engineers who demonstrate an ability to design and develop software in a professional manner, making use of Software Engineering knowledge, appropriate tools and practices.

- EO2: To strive to meet the specific needs of industry and/or academia, able to contribute effectively in research and development, as part of a team and on an individual basis.

- EO3: To be aware of the need for life long-learning and personal and professional growth in a computing world that is continually changing whilst possessing ethical consciousness and global awareness.

- EO4: To work towards demonstrating leadership and entrepreneurship in their profession.

The outcomes of the software engineering programme are essentially rooted to the needs and requirements of the software engineering sector, IT industry or academia. In addition, the development of the program outcomes strongly relied on feedback experts in the sector, experienced lecturers and previous graduates. Furthermore, our programme objectives are periodically refined via student-oriented surveys at the end of each semester and at the end of the programme. Specifically, we consider the expectations, needs and satisfaction of our students in relation to the programme which also has a measured impact on the further development of our programme outcomes and, moreover, the software engineering degree programme. Thus, the learning outcomes of the programme are established as in the following. Students graduating from the program should have:

- PO1: Adequate knowledge in mathematics, science and engineering subjects pertaining to the Software Engineering discipline; ability to use theoretical and applied knowledge to solve complex engineering problems.

- PO2: Ability to identify, formulate, and solve complex engineering problems; ability to select and apply proper analysis and modelling methods for this purpose.

- PO3: Ability to design a complex system, process, device or product under realistic constraints and conditions, in such a way as to meet the desired result; ability to apply modern design methods for this purpose.

- PO4: Ability to devise, select, and use modern techniques and tools needed for analyzing and solving complex problems encountered in Software Engineering practice; ability to employ information technologies effectively.

- PO5: Ability to design and conduct experiments, gather data, analyze and interpret results for investigating complex engineering problems or Software Engineering specific research questions.

- PO6: Ability to work efficiently in intra-disciplinary and multi-disciplinary teams; ability to work individually.

- PO7: Ability to communicate effectively in Turkish, both orally and in writing; knowledge of a minimum of one foreign language; ability to write effective reports and comprehend written reports, prepare design and production reports, make effective presentations, and give and receive clear and intelligible instructions.

- PO8: Recognition of the need for lifelong learning; ability to access information, to follow developments in science and technology, and to continue to educate him/herself.

- PO9: Consciousness to behave according to ethical principles and professional and ethical responsibility; knowledge on standards used in engineering practice.

- PO10: Knowledge about business life practices such as project management, risk management, and change management; awareness in entrepreneurship, innovation; knowledge about sustainable development.

- PO11: Knowledge about the global and social effects of Software engineering practices on health, environment, and safety, and contemporary issues of the century reflected into the field of engineering; awareness of the legal consequences of engineering solutions.

- PO12: Competence in modelling software architectures, using design patterns and implementing requirements analysis.

- PO13: Knowledge in software quality assurance activities and documentation together with testing strategies for producing engineering projects.

Curriculum

SOFTWARE ENGINEERING									
1-3-5-7 SEMESTER					2-4-6-8 SEMESTER				
COURSE CODE	COURSE NAME	CREDIT	ECTS	TYPE	COURSE CODE	COURSE NAME	CREDIT	ECTS	TYPE
COM101	ENGLISH I	(3,0)3	3	Compulsory	COM108 / ORT108	HISTORY / TARİH	(2,0)2	2	Compulsory
COM111	CHEMISTRY	(3,0)3	4	Compulsory	COM110	ENGLISH II	(3,0)3	3	Compulsory
ENG131	PHYSICS I	(3,0)3	4	Compulsory	COM122	PHYSICS II	(3,0)3	5	Compulsory
COMP117	COMPUTING FOUNDATIONS	(3,2)4	6	Compulsory	COMP124	COMPUTER PROGRAMMING	(3,2)4	6	Compulsory
COMP119	INTRODUCTION TO PROFESSION	(2,0)0	2	Compulsory	ENG122	PHYSICS II LAB	(0,2)1	2	Compulsory
ENG111	CHEMISTRY LAB	(0,2)1	2	Compulsory	MATH109	LINEAR ALGEBRA	(3,0)3	5	Compulsory
ENG121	PHYSICS I LAB	(0,2)1	2	Compulsory	MATH110	CALCULUS II	(3,2)4	7	Compulsory
MATH101	CALCULUS I	(3,2)4	7	Compulsory					
		19	30				20	30	
COMP205	DIGITAL LOGIC DESIGN	(3,2)4	6	Compulsory	COMP214	OPERATING SYSTEMS	(3,0)3	6	Compulsory
COMP209	DISCRETE MATHEMATICS	(3,0)3	5	Compulsory	COMP218	OBJECT-ORIENTED PROGRAMMING I	(3,2)4	8	Compulsory
COMP217	DATA STRUCTURES	(3,2)4	7	Compulsory	CFE202	ENVIRONMENT AND SUSTAINABLE DEVELOPMENT	(3,0)3	4	Free Elective
CFE201	LEADERSHIP AND MANAGEMENT	(3,0)3	4	Free Elective	MATH224	ENGINEERING MATHS	(3,0)3	5	Compulsory
MATH201	ORDINARY DIFFERENTIAL EQUATIONS	(3,2)4	5	Compulsory	MATH226	PROBABILITY & STATISTIC METHODS	(3,0)3	5	Compulsory
					SENG212	SOFTWARE REQUIREMENTS ANALYSIS AND SPECIFICATION	(3,0)3	5	Compulsory
		18	27				19	33	
COMP315	OBJECT-ORIENTED PROGRAMMING II	(3,2)4	8	Compulsory	COM106 / ORT106	TURKISH / TÜRKÇE	(2,0)2	2	Compulsory
COMP335	ANALYSIS OF ALGORITHMS	(3,0)3	4	Compulsory	COMP342	COMPUTER NETWORKS	(3,0)3	5	Compulsory
COMP337	DATABASE MANAGEMENT SYSTEMS	(3,2)4	7	Compulsory	COMP364	PRINCIPLES OF PROGRAMMING LANGUAGES	(3,2)4	7	Compulsory
COMP339	SYSTEMS PROGRAMMING	(3,0)3	4	Compulsory	SENG308	SOFTWARE QUALITY ASSURANCE AND TESTING	(3,0)3	5	Compulsory
SENG305	SOFTWARE DESIGN AND ARCHITECTURE	(3,0)3	7	Compulsory	SENG312	HUMAN COMPUTER INTERACTION	(3,0)3	6	Compulsory
					COMP471	JAVA PROGRAMMING	(3,0)3	5	Elective
		17	30				18	30	
ECON413	ENGINEERING ECONOMICS	(3,0)3	5	Compulsory	ENGG434	ENGINEERING ETHICS	(3,0)3	5	Compulsory
BUSN461	STRATEGIC PLANNING AND MANAGEMENT	(3,0)3	5	Compulsory	SENG450	GRADUATION PROJECT II	(0,6)3	10	Compulsory
SENG360	SUMMER TRAINING	(0,0)0	2	Compulsory	COMP464	INTERNET PROGRAMMING	(3,0)3	5	Elective
SENG407	SOFTWARE PROJECT MANAGEMENT	(3,0)3	5	Compulsory	COMP448	ARTIFICIAL NEURAL NETWORKS	(3,0)3	5	Elective
SENG451	GRADUATION PROJECT I	(0,3)1	3	Compulsory	EE431	PRINCIPLES OF DIGITAL IMAGE PROCESSING	(3,0)3	5	Elective
COMP415	ARTIFICIAL INTELLIGENCE	(3,0)3	5	Elective					
EE419	WIRELESS COMMUNICATION	(3,0)3	5	Elective					
		16	30				15	30	

DEPARTMENT OF
SOFTWARE ENGINEERING

COURSE CATALOGUE DESCRIPTIONS

Course Name	English I
Course Level	Undergraduate
Course Code	COM101
Semester	Fall
Person Responsible for the course	Lecturer Mehmet Mert
Lecturer	Lecturer Mehmet Mert
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 1st semester
Type of teaching, expected class size	Online, >500 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 3 hours per week 3. Total Exercises and Examination Preparation time: 40 hours
Credit Points - ECTS	3 Credit Points – 4 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	This course introduces the main grammatical structures to the students and helps them to develop their listening, speaking, reading and writing skills as well as vocabulary and pronunciation. The students are provided with clear rules and example sentences. The lessons contain high frequency vocabulary that the students are likely to come across during their studies and future their future careers.
Course Learning Outcomes	On successful completion of this course, students will have developed knowledge and understanding of: 1.The students will be able to understand and use English structures accurately to express themselves. 2. The students will be able to learn and use the vocabulary learnt during the lessons in real life contexts.
Study and examination requirements and forms of examination	• Midterm Examination • Final Examination
Media Employed	Projector, Moodle, Microsoft Teams
Reading List/ Recommended Text Book	Main: English File, Pre-Intermediate Plus, Student's Book, Christina Latham- Koenig, et al, Oxford University Press, Third Edition Supporting: English File, Pre-Intermediate Plus, Workbook, Christina Latham- Koenig, et al, Oxford University Press, Third Edition

Course Name	Chemistry
Course Level	Undergraduate
Course Code	COM111
Semester	Spring
Person Responsible for the course	Assist. Prof. Dr Saltuk Pirgalıoğlu
Lecturer	Assist. Prof. Dr Saltuk Pirgalıoğlu
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 1st semester
Type of teaching, expected class size	Face to face lectures, <100 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 3 hours per week 3. Total exercises and Examination Preparation time: 39 hours
Credit Points - ECTS	3 Credit Points – 4 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	
Catalogue Descriptions/Content	The aim of this course is to describe students how substances interact with one another. Students will be informed on how the atom is made up, how atoms come together to make molecules and how molecules can interact, chemical compounds, chemical bonds, chemical equations and reactions, aqueous solutions, periodic table, gases, the electronic structure of the atom and introduction to thermochemistry.
Course Learning Outcomes	On successful completion of the course, the student will be able to (1) understand and carryout calculations on properties of Substances (2) learn atomic structure and naming of compounds (3) learn mole concept, balancing equations, stoichiometry (4) carry out calculations on aqueous reactions (5) learn ideal gasses, gas mixtures and gas properties (6) understand electronic configurations and covalent bonding (7) learn basic principles of thermochemistry
Study and examination requirements and forms of examination	• Midterm Examination 2 • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture note sharing
Reading List/ Recommended Text Book	Main: 1. William L. Masterton, Cecile N. Hurley, Chemistry Principles and Reactions, 8th edition Cengage Learning, 2016 Supporting: 1. Raymond Chang and Kenneth Goldsby, General Chemistry: The Essential Concepts 7th Edition, 2014

Course Name	Physics I
Course Level	Undergraduate
Course Code	ENG131
Semester	Fall
Person Responsible for the course	Asst. Prof. Dr. Nemika Cellatoğlu
Lecturer	Asst. Prof. Dr. Nemika Cellatoğlu
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 1st semester
Type of teaching, expected class size	Face to Face, <150 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 4 hours per week 3. Total Exercises and Examination Preparation time: 30 hours
Credit Points - ECTS	3 Credit Points – 4 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	Introduction to properties of physical quantities. Properties of one dimensional motion, and classification of uniform and non-uniform one dimensional motion. Difference between vector quantities and scalar quantities. Summation, subtraction, and multiplication of vectors. Properties of two dimensional motion. Laws of Newton and principle of inertia. Application of second law of Newton to different mechanical systems, including circular motion. Work and energy, law of conservation of energy, and conservation of mechanical energy. Third law of Newton, and momentum. Difference between elastic and inelastic collision.
Course Learning Outcomes	On successful completion of this course, all students will have developed knowledge and understanding of: 1. an ability to translate, interpret and extrapolate important scientific models and laws governing classical mechanics, 2. an ability to demonstrate critical thinking and problem solving skills in the area of physics, 3. an ability to perform mathematical modelling of basic problems and establish their analytic solutions in field of classical mechanics, 4. an understanding of the connection of course material to engineering applications.
Study and examination requirements and forms of examination	• Quiz • Midterm Examination • Final Examination
Media Employed	Whiteboard, Projector, Moodle, Microsoft Teams
Reading List/ Recommended Text Book	Main: Serway, Jewett, Physics for Scientists and Engineers with Modern Physics (2018) Supporting: 1. Halliday and Resnick, Fundamentals of Physics 2. Sears and Freedman, University Physics 10/e

Course Name	Computing Foundations
Course Level	Undergraduate
Course Code	COMP 117
Semester	Fall
Person Responsible for the course	Assoc. Prof. Dr. Ezgi Deniz Ülker
Lecturer	Assoc. Prof. Dr. Ezgi Deniz Ülker
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 1st semester
Type of teaching, expected class size	Open Access, online <200 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 3 hours per week 3. Total Exercises and Examination Preparation time: 70 hours
Credit Points - ECTS	4 Credit Points – 6 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	Introduction to general problem-solving concepts, algorithms and applications. Computer terminology, units, number systems. Steps in problem-solving. Problem solution, pseudocode, algorithms, flowcharts, data types, control structures. History of computers and programming. A simple C program layout, syntax and rules. C language basics, native types, identifiers, declarations, variables, expressions, assignments. Basic console input and output functions. Operators, unary, binary, mathematical, relational, equality and logical, precedence and associativity rules, type conversions and casting. Statements, flow of control. Sequential structure. Selective structure, if-else statement. Repetitive structure, while loop, do-while loop. Tracing a C code.
Course Learning Outcomes	On successful completion of this course, all students will have developed knowledge and understanding of: 1. Able to understand programming and computing concepts 2. Ability to analyze a problem to extract requirements and constraints 3. Develop an ability to solve a simple and develop algorithms for complex problems 4. Ability to trace programs 5. Knowledge of C programming language
Study and examination requirements and forms of examination	• In class exercises • Midterm Examination • Laboratory Works • Final Examination
Media Employed	Moodle for Lecture video, lecture note sharing. MS Teams for online lectures
Reading List/ Recommended Text Book	Main:Maureen Sprankle, “Problem Solving and Programming Concepts”, Pearson Prentice Hall, 2006, ISBN: 0-13-119459-3 Supporting:Marshall Brain, “The Basics of C Programming”, 2013.

Course Name	Introduction to Profession
Course Level	Undergraduate
Course Code	Comp119
Semester	Spring
Person Responsible for the course	Asst. Prof. Dr. Vesile Evrim
Lecturer	Asst. Prof. Dr. Vesile Evrim
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsary, 1st semester
Type of teaching, expected class size	Face to Face < 100 Students
Workload	1. Lectures: 1 Lecture hours per week 2. Self-Study: 1 hours per week 3. Total Exercises and Examination Preparation time: 10 hours
Credit Points - ECTS	2 Credit Points – 2 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	Computer and software engineering professions. Curriculums, courses, course contents, course assessments, letter-grading system. Summer internships, graduation projects. Seminars on academic and/or professional activities.
Course Learning Outcomes	On successful completion of the course, the student will be knowledgeable on: 1. their profession, 2. the curriculum and course contents, 3. the course assessment, letter grading policies and grade-point-average calculations, 4. the summer internship and graduation project requirements, 5. Basics about the computers
Study and examination requirements and forms of examination	• Quiz • Project
Media Employed	Whiteboard, Projector and Moodle for Lecture note sharing
Reading List/ Recommended Text Book	Main: 1- Engineering Summer Training booklet 2- Shelly Cashman Vermaat, Discovering Computers Essentials, 1st. ed. 2018

Course Name	Chemistry Laboratory
Course Level	Undergraduate
Course Code	ENG111
Semester	Fall
Person Responsible for the course	Asst. Prof. Dr. Saltuk Pirgalıoğlu
Lecturer	Asst. Prof. Dr. Saltuk Pirgalıoğlu, Asst. Prof. Dr. Devrim Özdal
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 1st semester
Type of teaching, expected class size	Face to Face, <150 Students
Workload	1. Laboratory work: 2 hours per week 2. Self-Study: 1 hours per week 3. Total Exercises and Examination Preparation time: 3 hours
Credit Points - ECTS	1 Credit Points – 2 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the laboratories to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	This course has been specially designed as a intensive introduction to the techniques of experimental chemistry. Molarity, Solution preparation, Calculation of density, distillation, separation methods, precipitation reaction, acid-base titration, thermochemistry.
Course Learning Outcomes	LO1: Students will develop skills in collecting and managing data in order to express their results in a precise and reliable quantitative or qualitative form on lab. reports, LO2: Students will be use basic apparatus, apply experimental methodologies in the chemistry laboratory setting and Demonstrate the basic laboratory safety concepts, LO3: To gain ability to calculate concentration of solution and learn preparation of solution experimentally, LO4: To understand separation techniques and apply in laboratory, LO5: an understanding thermodynamics laws, enthalpy and free energy concepts, LO6: To observe precipitation reaction experimentally and calculate percentage yield of experiments, LO7: Defining acids and bases and conduct acid-base titration experiment, LO8: To calculate density of materials..
Study and examination requirements and forms of examination	• Laboratory Projects (6) • Midterm Examination • Final Examination
Media Employed	Whiteboard, Projector, Moodle, Microsoft Teams
Reading List/ Recommended Text Book	Main: European University of Lefke Chemistry Laboratory Man. (2017) Supporting: William L. Masterton, Cecile N. Hurley, Chemistry Principles and Reactions, Cengage Learning, 8th Edition (2016)

Course Name	Physics I Laboratory
Course Level	Undergraduate
Course Code	ENG121
Semester	Fall
Person Responsible for the course	Asst. Prof. Dr. Nemika Cellatoğlu
Lecturer	Asst. Prof. Dr. Nemika Cellatoğlu
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 1st semester
Type of teaching, expected class size	Face to Face, <150 Students (Group size 30)
Workload	1. Laboratory work: 2 hours per week 2. Self-Study: 1 hours per week 3. Total Exercises and Examination Preparation time: 3 hours
Credit Points - ECTS	1 Credit Points – 2 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the laboratories to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	This course is directed to ENG131-Physics I. The aim of the course is providing a medium for students to see the experimental applications of kinematics and dynamics of one dimensional, two dimensional, circular and rotational motion. The course supports students to validate the underlying theory through experiment and observation.
Course Learning Outcomes	On successful completion of this course, all students will have developed knowledge and understanding of: 1. nature of science and scientific method 2. the ability to apply knowledge/skills to real world settings by identifying possible sources of error and implementing techniques that enhance precision. 3. an ability to demonstrate critical thinking and problem solving skills in the area of physics, 4. teamwork skills/ ability to collaborate by working in groups on a laboratory experiment. 5. written communication ability by reporting verbally the experimental data, results, and assessment of reliability.
Study and examination requirements and forms of examination	• Laboratory Projects (6) • Midterm Examination • Final Examination
Media Employed	Whiteboard, Projector, Moodle, Microsoft Teams
Reading List/ Recommended Text Book	Main: European University of Lefke Physics Laboratory Manual (2017) Supporting: Serway, Physics for Scientists and Engineers with Modern Physics 9/e, Cengage Learning

Course Name	Calculus I
Course Level	Undergraduate
Course Code	MATH101
Semester	Fall
Person Responsible for the course	Assoc. Prof. Dr. Yönal Kırsal
Lecturer	Assoc. Prof. Dr. Yönal Kırsal
Language	English
Relation to Curriculum	The undergraduate degree program, Compulsory, 1st semester
Type of teaching, expected class size	Face-to-face lectures, <250 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 3 hours per week 3. Total Exercises and Examination Preparation time: 80 hours
Credit Points - ECTS	4 Credit Points – 7 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	None
Catalogue Descriptions/Content	Fundamentals of calculus and its applications for engineers. The conceptual and visual representation of limits, continuity, differentiability, and tangent line approximations for functions at a point. Applying the power rule, product rule, quotient rule and chain rule to functions explicitly and implicitly for finding derivatives. Applying the fundamental theorem of calculus to evaluate definite integrals. Performing accurately improper integrals, definite and indefinite integration, integration by parts, substitution, and inverse trigonometric substitution.
Course Learning Outcomes	On successful completion of the course, the student should: (1) learn the Cartesian coordinates system, understand function evaluation, graph functions, recall composite functions, odd-even functions, the domain-range concept of the functions, and trigonometric functions; (2) understand conceptual and visual representation of limits, continuity, differentiability, and tangent line approximations for functions at a point; (3) apply the power rule, product rule, quotient rule and chain rule to functions explicitly and implicitly for finding derivatives; (4) apply the Fundamental Theorem of calculus to evaluate definite integrals, and calculate the area between the curves; (5) perform accurate substitution method, improper integrals, integration by parts, and inverse substitution.
Study and examination requirements and forms of examination	• In-class exercises • Midterm Examination 1 • Midterm Examination 2 • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture note sharing
Reading List/ Recommended Text Book	Main: Calculus: A Complete Course 8th Edition (2013), Robert A. Adams, Christopher Essex Supporting: Calculus 8th Edition (2016), James Stewart

Course Name	History
Course Level	Undergraduate
Course Code	COM108
Semester	Spring
Person Responsible for the course	Assoc. Prof. Dr. Osman Erciyas
Lecturer	Assoc. Prof. Dr. Osman Erciyas
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 2nd semester
Type of teaching, expected class size	Online, >500 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 2 hours per week 3. Total Exercises and Examination Preparation time: 38 hours
Credit Points - ECTS	2 Credit Points – 2 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	In this course the students who have been studying at different departments of our university will learn how the Ottoman Empire collapsed and a new Turkish Republic was found in the early 20th century. At the same time the students will learn the Eastern Question, Armenian Question and Cyprus Question which were created by different policy powers in the historical period. By the end of the semester the students will be able to understand why Mustafa Kemal is an important figure in the history of Turkey and the world. Besides, they will learn the Turkish Revolution and the establishment philosophy of the Turkish Republic and the principles of Mustafa Kemal.
Course Learning Outcomes	On successful completion of this course, students will learn: LO1: How the Ottoman Empire collapsed and a new Turkish Republic was found in the early 20th century, LO2: The Eastern Problem, Armenian Problem and Cyprus Problem, which were created by different policy powers in the historical period, LO3: Understand why Mustafa Kemal is an important figure in the history of Turkey and the World, LO4: The Turkish Revolution and the establishment philosophy of the Turkish Republic and the principles of Mustafa Kemal.
Study and examination requirements and forms of examination	• Midterm Examination • Final Examination
Media Employed	Projector, Moodle, Microsoft Teams
Reading List/ Recommended Text Book	Main: Supporting:

Course Name	English II
Course Level	Undergraduate
Course Code	COM110
Semester	Spring
Person Responsible for the course	Lecturer Mehmet Mert
Lecturer	Lecturer Mehmet Mert
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 2nd semester
Type of teaching, expected class size	Online, >500 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 3 hours per week 3. Total Exercises and Examination Preparation time: 40 hours
Credit Points - ECTS	3 Credit Points – 4 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	COM101 – English I
Catalogue Descriptions/Content	This course introduces the main grammatical structures to the students and helps them to develop their listening, speaking, reading and writing skills as well as vocabulary and pronunciation. The students are provided with clear rules and example sentences. The lessons contain high frequency vocabulary that the students are likely to come across during their studies and future their future careers.
Course Learning Outcomes	On successful completion of this course, students will have developed knowledge and understanding of: 1.The students will be able to understand and use English structures accurately to express themselves. 2. The students will be able to learn and use the vocabulary learnt during the lessons in real life contexts.
Study and examination requirements and forms of examination	• Midterm Examination • Final Examination
Media Employed	Projector, Moodle, Microsoft Teams
Reading List/ Recommended Text Book	Main: English File, Intermediate Plus, Student's Book, Christina Latham-Koenig, et al, Oxford University Press, Third Edition Supporting: English File, Intermediate Plus, Workbook, Christina Latham- Koenig, et al, Oxford University Press, Third Edition

Course Name	Physics II
Course Level	Undergraduate
Course Code	COM122
Semester	Spring
Person Responsible for the course	Asst. Prof. Dr. Nemika Cellatoğlu
Lecturer	Asst. Prof. Dr. Nemika Cellatoğlu
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 2nd semester
Type of teaching, expected class size	Face to Face, <150 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 4 hours per week 3. Total Exercises and Examination Preparation time: 30 hours
Credit Points - ECTS	3 Credit Points – 4 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	This course aims to introduce fundamental concepts of physics for engineering science and to provide essential background for engineering students. The course provides deep understanding of thermodynamics, electricity and magnetism. Also, the course aims to show the students the engineering applications of the course material.
Course Learning Outcomes	On successful completion of this course, all students will have developed knowledge and understanding of: 1. An ability to translate, interpret and extrapolate important scientific models and laws governing thermodynamics, electricity and magnetism. 2. An ability to demonstrate critical thinking and problem solving skills in the area of physics 3. An ability to perform mathematical modelling of basic problems and establish their analytic solutions in field of thermodynamics, electric and magnetism 4. An understanding of the connection of course material to engineering applications.
Study and examination requirements and forms of examination	• Quiz • Midterm Examination • Final Examination
Media Employed	Whiteboard, Projector, Moodle, Microsoft Teams
Reading List/ Recommended Text Book	Main: Serway, Jewett, Physics for Scientists and Engineers with Modern Physics (2018) Supporting: Halliday and Resnick, Fundamentals of Physics Sears and Freedman, University Physics 10/e Gettys, Keller and SKove, Physics: Classical and Modern

Course Name	Computer Programming
Course Level	Undergraduate
Course Code	COMP 124
Semester	Fall
Person Responsible for the course	Assist. Prof. Dr. Ferhun Yorgancıoğlu
Lecturer	Assist. Prof. Dr. Ferhun Yorgancıoğlu
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 2nd semester
Type of teaching, expected class size	Face to face lectures, <65 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 3 hours per week 3. Total Exercises and Examination Preparation time: 20 hours
Credit Points - ECTS	4 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	Computing Foundations
Catalogue Descriptions/Content	Overview of C language. Expressions: constants, data types, type modifiers, const qualifier, operators, order of evaluations, type conversions and casts. Types of statements: sequential, selective and iterative. Selective statements: if-else, switch-case, conditional operator, nested forms. Iterative statements: for, while, do-while, infinite loops, comma operator, break and continue statements, nested forms. Functions: definitions, formal parameters, actual arguments, function calls, call-by-value parameter passing, function prototypes, scope rules and storage classes. Arrays: declarations, initialisation lists, define directive, arrays as function arguments, call-by-reference parameter passing. Strings: character arrays, null character, string-handling functions. Pointers: declarations, pointer and array relationship, pointer arithmetic, array-subscript and pointer-offset notations, pointers as function arguments. Structures: user-defined types, typedef definitions, structures as function arguments, array of structures, pointer to a structure.
Course Learning Outcomes	Upon successful completion of the course, the student should be able to: 1. recall basics of the C language such as the data types, operators, expressions and order of evaluations 2. use if and switch statements to implement selective structures in C 3. use while, for and do-while loops to construct iterative structures in C 4. define and write functions in C to gain procedural programming skills 5. understand the scope rules and storage classes of C 6. use arrays and pointers in C and understand the close relationship between arrays and pointers 7. use strings in C and define and write user-defined string-handling functions 8. use structures in C and understand the concept of a user-defined data type
Study and examination requirements and forms of examination	• In class exercises • Laboratory works • Midterm Examination • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture note sharing
Reading List/ Recommended Text Book	Main: S.G. Kochan, Programming in C, 4th ed., Addison-Wesley, 2015. Supporting: S. Prata, C Primer Plus, 6th ed., Addison-Wesley, 2014.

Course Name	Physics II Laboratory
Course Level	Undergraduate
Course Code	ENG122
Semester	Spring
Person Responsible for the course	Mehmet Burhan
Lecturer	Mehmet Burhan
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 2nd semester
Type of teaching, expected class size	Face to Face, <150 Students (Group size 30)
Workload	1. Laboratory work: 2 hours per week 2. Self-Study: 1 hours per week 3. Total Exercises and Examination Preparation time: 3 hours
Credit Points - ECTS	1 Credit Points – 2 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the laboratories to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	This course is directed with COM122-Physics II. The aim of course is providing a medium for students to see the experimental applications of thermodynamics, electricity and magnetism. The course supports students to validate the underlying theory through experiment and observation.
Course Learning Outcomes	On successful completion of this course, all students will have developed knowledge and understanding of: 1. Conduct experimental investigations of simple electric, magnetic and thermodynamic phenomena. 2. Carry out measurements utilizing appropriate techniques and safety practices. 3. Practice record keeping of experimental work and data graphing. 4. Analyze data using simple statistics and compare the results with the relevant theory 5. Write a lab report including a summary explaining the theoretical background and major experimental achievements and findings..
Study and examination requirements and forms of examination	• Laboratory Projects (6) • Midterm Examination • Final Examination
Media Employed	Whiteboard, Projector, Moodle, Microsoft Teams
Reading List/ Recommended Text Book	Main: European University of Lefke Physics II Laboratory Booklet Supporting: Serway, Physics for Scientists and Engineers with Modern Physics 9/e, Cengage Learning

Course Name	Linear Algebra
Course Level	Undergraduate
Course Code	MATH109
Semester	Spring
Person Responsible for the course	Asst. Prof. Dr. Ferhun Yorgancıoğlu
Lecturer	Asst. Prof. Dr. Ferhun Yorgancıoğlu
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 2nd semester
Type of teaching, expected class size	Face to Face, <150 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 6 hours per week 3. Total Exercises and Examination Preparation time: 30 hours
Credit Points - ECTS	3 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	Systems of linear equations. Elementary row operations, echelon forms, Gaussian elimination. Matrices, power of matrices, determinants, inverses, diagonal matrices. Cofactor expansion via row reduction. Cramer's rule and evaluating determinants. Vector spaces, linear independence, basis, dimension, inner product spaces, Euclidean spaces. Linear transformation systems. Eigenvalues and eigenvectors; and eigenvalue/eigenvector applications.
Course Learning Outcomes	On successful completion of this course, all students will have developed knowledge and understanding of: 1. some rudimentary applications of systems of linear equations, 2. calculation of determinants using row operations, column operations and expansion down any column and across any row, 3. performing the operations of addition, scalar multiplication and find the transpose and inverse of a matrix, 4. proving statements of an algebraic nature concerning linear transformations, 5. calculating eigenvalues and their corresponding eigenspaces.
Study and examination requirements and forms of examination	• Quiz • Midterm Examination • Final Examination
Media Employed	Whiteboard, Projector, Moodle, Microsoft Teams
Reading List/ Recommended Text Book	Main: Gilbert Strang, Introduction to Linear Algebra, 5th ed., Wellesley - Cambridge Press, (2016). Supporting: Seymour Lipschutz, Theory and Problems of Linear Algebra, 2nd ed., Shaum's Outline Series, McGraw-Hill (1991).

Course Name	Calculus II
Course Level	Undergraduate
Course Code	MATH110
Semester	Spring
Person Responsible for the course	Assoc. Prof. Dr. Yönal Kırsal
Lecturer	Assoc. Prof. Dr. Yönal Kırsal
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 2nd semester
Type of teaching, expected class size	Face to Face, <150 Students
Workload	1. Lectures: 4 Lecture hours per week 2. Self-Study: 8 hours per week 3. Total Exercises and Examination Preparation time: 44 hours
Credit Points - ECTS	4 Credit Points – 7 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	Techniques of integration, integration by parts, trigonometric substitution, integration of rational functions, integration of trigonometric integrals. Application of integrals, areas between curves, volume, volumes by slicing, volumes by cylindrical shells, arc length, area of a surface of revolution, moments and centre of mass. Parametric equations, curves defined by parametric equations, calculus with parametric equations, derivation, area and arc length calculations. Polar coordinates, plotting with polar coordinates, derivation and integration with polar coordinates. Sequences, series, integral tests and estimates of sum.
Course Learning Outcomes	On successful completion of the course, the student will be able to: 1.evaluate an integral by the method of substitution; 2. use integrals to calculate areas between curves, volumes, work, and average value of a function; 3. evaluate integrals, using the techniques of integration by parts, using trigonometric identities and trigonometric substitution, and using partial fractions; 4. evaluate the two types of improper integrals; 5. use integrals to find arc length and area of a surface of revolution; 6. use integrals in applications to physics and engineering; 7. describe curves in parametric form and polar coordinates; 8. determine whether or not a sequence of real numbers converges; 9. test a series for convergence or divergence, using the integral, ratio, root, and comparison tests.
Study and examination requirements and forms of examination	• Midterm Examination • Final Examination
Media Employed	Whiteboard, Projector, Moodle, Microsoft Teams
Reading List/ Recommended Text Book	Main: James Stewart, Calculus, 8th Ed. Cengage, 2015 (Textbook) Supporting: 1. Robert A. Adams, Christopher Essex, Calculus: A Complete Course, 9th Ed., Pearson Education Inc., 2017. (Reference). George B. Thomas, Maurice D. Weir, Joel R. Hass, Thomas' Calculus, 14th Ed., Pearson Education Inc., 2017. (Reference)

Course Name	Digital Logic Design
Course Level	Undergraduate
Course Code	COMP205
Semester	Fall
Person Responsible for the course	Asst. Prof. Dr. Ahmet Yaşlı
Lecturer	Asst. Prof. Dr. Ahmet Yaşlı
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 3rd semester
Type of teaching, expected class size	Face to Face, <150 Students
Workload	1. Lectures: 2 Lecture hours per week 2. Self-Study: 6 hours per week 3. Laboratories: 2 hours per week 4. Total Exercises and Examination Preparation time: 22 hours
Credit Points - ECTS	4 Credit Points – 6 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	Number systems; logic gates; Boolean algebra and truth tables; Karnaugh maps; Combinational logic design; Decoders and encoders; Multiplexers; Arithmetic circuits; Synchronous sequential circuits; Flip-flops; Counters and shift registers; Sequential logic design; Memory and programmable logic.
Course Learning Outcomes	On successful completion of this course, all students will have developed knowledge and understanding of: 1.different number systems 2.procedures/techniques for simplifying Boolean functions 3.analysis/design of basic combinational logic circuits 4.analysis/design of basic clocked sequential logic circuits.
Study and examination requirements and forms of examination	• Laboratory Work (7 laboratory projects) • Midterm Examination • Final Examination
Media Employed	Whiteboard, Projector, Moodle, Microsoft Teams
Reading List/ Recommended Text Book	Main: J. F. Wakerly, Digital Design: Principles and Practices, 4th Ed, Prentice Hall, 2019. ISBN-10: 013446009X (Textbook) Supporting: M. Morris Mano, Digital Design, 5th Ed, Prentice Hall, 2012. ISBN-10: 0-13-277420-

Course Name	Discrete Mathematics
Course Level	Undergraduate
Course Code	COMP209
Semester	Fall
Person Responsible for the course	Asst. Prof. Dr. Zafer Erenel
Lecturer	Asst. Prof. Dr. Zafer Erenel
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 3rd semester
Type of teaching, expected class size	Face to Face, <150 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 6 hours per week 3. Total Exercises and Examination Preparation time: 35 hours
Credit Points - ECTS	3 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	The main purpose of this course is to introduce the concepts of discrete math. The subjects include logical forms, logical equivalences, conditional statements, digital logical circuits and number systems, algorithms, sequences and mathematical induction. The emphasis is placed on recursion. This course also covers set theory, counting and probability theory, functions, relations and graphs.
Course Learning Outcomes	On successful completion of this course, all students will have developed knowledge and understanding of: 1. An ability to construct simple mathematical proofs 2.To become familiar with the basic principles of Boolean algebra, Logic, Set theory 3. An understanding of mathematical concepts and terminology 4. An ability to apply algorithmic thinking in problem solving.
Study and examination requirements and forms of examination	• Midterm Examination • Final Examination
Media Employed	Whiteboard, Projector, Moodle, Microsoft Teams
Reading List/ Recommended Text Book	Main: Epp,S.S.; Discrete Mathematics with Applications 4th ed. , Brooks/Cole Cengage Learning, 2010 (Textbook) Supporting: Rosen, K.H.; Discrete Mathematics and Its Applications 6th ed. , McGraw-Hill, 2007

Course Name	Data Structures
Course Level	Undergraduate
Course Code	COMP 217
Semester	Fall
Person Responsible for the course	Assist. Prof. Dr. Ferhun Yorgancıoğlu
Lecturer	Assist. Prof. Dr. Ferhun Yorgancıoğlu
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 3rd semester
Type of teaching, expected class size	Face to face lectures, <65 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 3 hours per week 3. Total Exercises and Examination Preparation time: 50 hours
Credit Points - ECTS	4 Credit Points – 7 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	Computer Programming
Catalogue Descriptions/Content	Notion of data and types, representations in computer memory, abstract data type, types of data structures. The stack structure: properties, operations, types of problems solved, array-based implementations, case study: shunting-yard algorithm. The queue structure: properties, operations, types of problems solved, array-based implementations, circular queue, case study: waiting lines. The linked list structure: singly, doubly, and circular lists, basic operations and applications, linked stack and linked queue. Recursion: recursive definitions, how to write a recursive C function, working on examples, case study: linear vs binary search. The tree structure: definitions and properties, binary trees, tree traversals, binary search trees, inserting and deleting nodes into/from a binary search tree.
Course Learning Outcomes	Upon successful completion of the course, the student should have gained: 1. knowledge of data and its types, their memory representations, and the abstract data type 2. theoretical knowledge and practical skills on how to solve computing problems using the stack structure 3. theoretical knowledge and practical skills on how to solve computing problems using the queue structure 4. theoretical knowledge of recursion and the ability to define and implement recursive functions 5. theoretical knowledge and programming experience on linked lists 6. theoretical knowledge and programming experience on binary search trees
Study and examination requirements and forms of examination	• In class exercises • Laboratory works • Programming assignments • Midterm Examination • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture note sharing
Reading List/ Recommended Text Book	Main:Y. Langsam, M.J. Augenstein, A.M. Tenenbaum, Data Structures using C and C++, 2nd ed., Prentice Hall, 1996. Supporting: M.T. Goodrich, R. Tamassia, D.M. Mount, Data Structures and Algorithms in C++, 2nd ed., John Wiley, 2011.

Course Name	Leadership and Management
Course Level	Undergraduate
Course Code	CFE201
Semester	Fall
Person Responsible for the course	Asst. Prof. Dr. Tahir Yeşilada
Lecturer	Asst. Prof. Dr. Tahir Yeşilada
Language	English
Relation to Curriculum	Undergraduate degree program, Elective, 3rd semester
Type of teaching, expected class size	Face to Face, <150 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 1 hours per week 3. Total Exercises and Examination Preparation time: 50 hours
Credit Points - ECTS	3 Credit Points – 4 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	In this course, an analysis of theoretical and practical knowledge is made. In this context, basic social and psychological factors associated with the concept of leadership and current theories will be explained and how theoretical knowledge can be applied in terms of leadership and management functions in organizations will be emphasized. The aim of the course is to provide students with a deep understanding of leadership and management concepts and to develop their own leadership skills.
Course Learning Outcomes	At the end of this course, the student will be able to: 1. Describe the concept of leadership and effective leadership theories, 2. Distinguish between managerial processes, traits & skills, 3. Recall diversity and the challenges of workforce environments, 4. Identify the effective use of power & influence in organizations, 5. Recognize leading change in organizations.
Study and examination requirements and forms of examination	• Midterm Examination • Final Examination
Media Employed	Whiteboard, Projector, Moodle, Microsoft Teams
Reading List/ Recommended Text Book	Main: Leadership in organizations, Gary Yuki and William Gardner, Pearson Education (2020) (Textbook)

Course Name	Ordinary Differential Equations
Course Level	Undergraduate
Course Code	MATH201
Semester	Fall
Person Responsible for the course	Assist. Prof. Dr Salih Karanfil
Lecturer	Assist. Prof. Dr Salih Karanfil
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 4th semester
Type of teaching, expected class size	Face to face lectures, <65 Students
Workload	1. Lectures: 4 Lecture hours per week 2. Self-Study: 3 hours per week 3. Total Exercises and Examination Preparation time: 30 hours
Credit Points - ECTS	4 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	Calculus 1
Catalogue Descriptions/Content	Definition and classification of differential equations. Solution of first order linear differential equations, initial value problems, homogeneous differential equations, non-homogeneous differential equations, separation of variables, exact differential equations, integrating factors, the method of undetermined coefficient, Bernoulli equations, higher order differential equations, Systems of linear differential equations with constant coefficients, Cauchy Euler equations, Laplace transforms and properties of Laplace Transforms.
Course Learning Outcomes	On successful completion of this course, all students will have developed knowledge and understanding of: 1. Identify types of differential equations 2. Solving first order differential equations 3. Solving higher order differential equations 4. Solving Inhomogeneous linear systems 5. Basics of Laplace Transforms
Study and examination requirements and forms of examination	• In class exercises • Midterm Examination 1 • Midterm Examination 2 • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture note sharing
Reading List/ Recommended Text Book	Main: S.L ROSS, Introduction to Ordinary Differential Equations, 4th Edition, John Wiley & Sons, 1989 Supporting: Polking, Bogges, Arnold, “Differential Equations”, Pearson Education, 2006 EDWARDS & PENNY, Elementary Differential Equations, 6th Edition Pearson Education, 2013.

Course Name	Operating Systems
Course Level	Undergraduate
Course Code	COMP214
Semester	Spring
Person Responsible for the course	Assist. Prof. Dr. Zafer Erenel
Lecturer	Assist. Prof. Dr. Zafer Erenel
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 4th semester
Type of teaching, expected class size	Face to face lectures, <100 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 3 hours per week 3. Total Exercises and Examination Preparation time: 30 hours
Credit Points - ECTS	3 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	
Catalogue Descriptions/Content	The history of the operating systems. The hardware and software components. Application Programming Interface. Computing environments. Process Management,Process operations, Process synchronization. Processor Scheduling Criteria and Algorithms (FIFO, RR, SJF, SRTF, PRIORITY, PREEMPTIVE, NONPREEMPTIVE ALGORITHMS). The Indefinite postponement, Deadlock prevention, detection, avoidance, recovery. The main memory, swapping, fixed partition,multiprogramming, variable partition multiprogramming, paging, segmentation. The virtual memory, page replacement strategies (FIFO,LRU,OPT). Secondary Storage, Disk scheduling(First-Come-First-Served, Shortest-Seek-Time-First, SCAN, C-SCAN, FSCAN, N-Step SCAN,LOOK, C-LOOK). File Systems, Directories.
Course Learning Outcomes	On successful completion of this course, all students will have developed knowledge and understanding of: (1) knowledge of the main differences between parallel, distributed, real-time and hand-held systems. (2) an ability to implement deadlock avoidance, prevention and detection algorithms. (3) an ability to implement memory management algorithms for swapping, paging and virtual memory. (4) an ability to implement process scheduling algorithms. (5) an ability to implement disk scheduling algorithms. (6) competence in using DOS commands and UNIX commands.
Study and examination requirements and forms of examination	• In class examples • Midterm Examination • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture note sharing
Reading List/ Recommended Text Book	Main: (1) H.M. Dietel, P.J. Dietel, D.R. Choffnes, Operating Systems, 3rd ed., Pearson, 2004 Supporting: (1) A. Silberschatz, P.B. Galvin, G. Gagne, Operating System Concepts, 9th ed., Wiley, 2012 (2) A.S. Tanenbaum, H. Bos, Modern Operating Systems, Pearson, 4th ed., 2016

Course Name	Object-Oriented Programming
Course Level	Undergraduate
Course Code	COMP 218
Semester	Fall
Person Responsible for the course	Assist. Prof. Dr. Ferhun Yorgancıoğlu
Lecturer	Assist. Prof. Dr. Ferhun Yorgancıoğlu
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 4th semester
Type of teaching, expected class size	Face to face lectures, <65 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 3 hours per week 3. Total Exercises and Examination Preparation time: 60 hours
Credit Points - ECTS	4 Credit Points – 8 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	Computer Programming
Catalogue Descriptions/Content	Overview of C++ language. Console input and output streams of C++. C++ functions: inline functions, function overloading, default arguments and reference parameters. C++ operators new and delete. C++ vectors. C++ strings. Classes and objects: abstraction, data members and member functions, encapsulation, hiding data, member access control, accessor and mutator functions, constructors, destructor, constant and static data members and member functions, friend functions, friend classes, object composition, copy constructor, member initializer syntax, this pointer, cascaded function calls. Operator overloading: overloading as member, non-member and friend functions. Inheritance: base and derived classes, protected access, private inheritance, multilevel inheritance, multiple base classes, virtual base class. Polymorphism and virtual functions: polymorphic functions, static vs dynamic binding, pure virtual functions, abstract classes, virtual destructor.
Course Learning Outcomes	Upon successful completion of the course, the student should be able to: 1. carry structured programming skills previously gained in C language to C++ 2. obtain programming experience on C++ vector and string objects 3. build a strong foundation about “data abstraction” and “data hiding” notions through the concept of encapsulation 4. obtain knowledge and practice of “code reuse” principle through the concepts of inheritance and composition 5. obtain knowledge and practice of “one interface, multiple methods” principle through the concept of polymorphism
Study and examination requirements and forms of examination	• In class exercises • Laboratory works • Programming assignments • Midterm Examination • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture note sharing
Reading List/ Recommended Text Book	Main: H.M. Deitel, P.J. Deitel, C++ How to Program, 10th ed., Pearson, 2017. Supporting: S. Prata, C++ Primer Plus, 6th ed., Addison-Wesley, 2011.

Course Name	Environmental and Sustainable Development
Course Level	Undergraduate
Course Code	CFE202
Semester	Spring
Person Responsible for the course	Assist. Prof. Dr Semih Oğuzcan
Lecturer	Assist. Prof. Dr Semih Oğuzcan
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 4th semester
Type of teaching, expected class size	Online lectures, >65 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 4 hours per week 3. Total Exercises and Examination Preparation time: 33 hours
Credit Points - ECTS	3 Credit Points – 4 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	Ecology and sustainability, biodiversity, urbanization, ecological succession, climate and biodiversity, sustaining biodiversity, sustaining resources and environmental quality: food production, water resources and pollution, mineral sources, energy sources, environmental hazards and human health, air pollution, ozone depletion, climate change, solid and hazardous wastes
Course Learning Outcomes	On successful completion of the course, the student will be able to: (1) understand environmental problems (2) construct relationship between ecology, biodiversity and sustainability (3) apply principles of sustainability on various environmental issues (4) understand sustainable resources management: water, energy, minerals (5) understand how resource consumption affect air pollution and climate change (6) understand advantages and disadvantages of different strategies in solid waste management
Study and examination requirements and forms of examination	• Midterm Examination 1 • Final Examination • Project 1
Media Employed	Whiteboard, Projector and Moodle for Lecture note sharing
Reading List/ Recommended Text Book	Main: 1. G. Tyler Miller, Scott Spoolman,«Environmental Science», 15th Edition, Cengage Learning, 2016 (Textbook) Supporting: Jefferson W. Tester, Elisabeth M. Drake, Michael J. Driscoll, Michael W. Golay, and William A. Peters, «Sustainable Energy Choosing Among Options»,2nd Edition, MIT Press, 2012.

Course Name	Engineering Mathematics
Course Level	Undergraduate
Course Code	MATH224
Semester	Spring
Person Responsible for the course	Assist. Prof. Dr Saltuk Pirgalıoğlu
Lecturer	Assist. Prof. Dr Saltuk Pirgalıoğlu
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 4th semester
Type of teaching, expected class size	Face to face lectures, <100 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 4 hours per week 3. Total Homework, exercises and Examination Preparation time: 45 hours
Credit Points - ECTS	3 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	Calculus I
Catalogue Descriptions/Content	The concept of numerical error, solution of nonlinear equations with root finding. Solution of linear systems of equations using software packages. Direct and iterative methods for the solution of linear algebraic equations. Polynomial interpolation (Lagrange and Newton polynomials) and extrapolation. Curve fitting for least squares line and polynomial fitting with data linearization method. Numerical differentiation, numerical integration with quadrature formulas and their error analysis. Numerical solution of ordinary differential equations.
Course Learning Outcomes	On successful completion of the course, students should have gained: (1) an ability to recognize the difference between analytical and numerical solutions (roundoff and truncation errors), (2) knowledge of bracketing and open methods to solve root of equation problems, (3) an ability to solve simultaneously sets of linear algebraic equations using Naive Gauss Elimination, (4) ability to differentiate the fundamental difference between regression and interpolation and to solve the numerical method problems, (5) ability to solve numerical differentiation, ordinary differential equations and integration problems. (6) Having knowledge of complex numbers (7) ability to solve optimization problems
Study and examination requirements and forms of examination	• Homework • Midterm Examination 2 • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture note sharing
Reading List/ Recommended Text Book	Main: S.C.Chapra, Raymond P Canale, Numerical Methods for Engineers, 7th Edition, McGrawHill, 2015 Supporting: John. H. Mathews, Kurtis D. Fink, Numerical Methods Using MATLAB, 4th Edition Pearson Prentice Hall, 2004

Course Name	Probability and Statistic Methods
Course Level	Undergraduate
Course Code	MATH226
Semester	Spring
Person Responsible for the course	Assist. Prof. Dr Semih Oğuzcan
Lecturer	Assist. Prof. Dr Semih Oğuzcan
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 2nd semester
Type of teaching, expected class size	Face to face lectures, >65 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 3 hours per week 3. Total Exercises and Examination Preparation time: 35 hours
Credit Points - ECTS	3 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	Basic probability laws, Random variables, Operations on Random Variables, Multiple random variables, Operations on multiple random variables, Sampling distributions, distribution functions, continuous distribution functions, Gaussian distribution and its properties and applications, Gaussian Q-function and its applications, Important discrete distributions, Simple discrete distributions (binomial, hypergeometric, combination, permutation)
Course Learning Outcomes	On successful completion of this course, all students will have developed knowledge and understanding of: 1) Understand basic concepts in probability including combinatorics, independence, conditional probability and Bayes rule. 2) Compute probabilities by modelling sample spaces and applying rules of permutations and combinations, additive and multiplicative laws and conditional probability. 3) Solve basic problems arising in engineering that involve discrete and continuous probability distributions. 4) Construct the probability distribution of a random variable, based on a real-world situation, and use it to compute expectation and variance. 5) Use statistical concepts such as means, variances and various types of graphs to analyse datasets, and sampling distributions.
Study and examination requirements and forms of examination	• Midterm Examination 1 • Final Examination • Quiz 1
Media Employed	Whiteboard, Projector and Moodle for Lecture note sharing
Reading List/ Recommended Text Book	Main: Sheldon Ross, "Introduction to Probability and Statistics for Engineers and Scientists", Elsevier, Academic press, ISBN 10: 0-12-370483-9. Supporting: R. E. Walpole, R. H. Myers, S. L. Myers and K. E. Ye, "Probability & Statistics for Engineers and Scientists", Prentice-Hall, 2011.

Course Name	Software Requirements Analysis and Specifications
Course Level	Undergraduate
Course Code	SENG 212-303
Semester	Spring
Person Responsible for the course	Assoc. Prof. Dr. Ezgi Deniz Ülker
Lecturer	Assoc. Prof. Dr. Ezgi Deniz Ülker
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 4th semester
Type of teaching, expected class size	Open Access, online <80 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 4 hours per week 3. Total Exercises and Examination Preparation time: 50 hours
Credit Points - ECTS	3 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	Introduce important software properties; security, maintenance, adaptability, robustness, safety, reliability, emergent properties, non-emergent properties. Software process steps; requirement analysis, specification, prototype, design, implementation, testing, validation and verification. Process models; waterfall, evolutionary, incremental, spiral component based, agile processes, extreme programming, pair programming. Functional and non-functional requirements, system requirements, domain requirements, external requirements, open and closed interview, external requirements, constraints, Organizational Goals, Project Goals, Stakeholders, Life Cycle Planning, responsibilities, activity network, product feasibility, organizational feasibility, financial feasibility, use case diagram, scenarios.
Course Learning Outcomes	On successful completion of this course, all students will have developed knowledge and understanding of: 1. learn system attributes and management of the problems in systems engineering 2. learn managing the risk and developing project plan, 3. learn formal project documentation (SRS), 4. learn about Requirement Engineering and software processes, 5. Improve communication skills as a member of a team
Study and examination requirements and forms of examination	• In class exercises • Midterm Examination • Assignments • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture video links, lecture note sharing. MS Teams for recording the lectures.
Reading List/ Recommended Text Book	Main: Ian Sommerville, Software Engineering, 10th Edition, Addison Wesley, 2015. Supporting: http://sunset.usc.edu/~neno/cs477_2003/MBASE_Guidelines.doc

Course Name	Object Oriented Programming II
Course Level	Undergraduate
Course Code	COMP315
Semester	Fall
Person Responsible for the course	Asst. Prof. Dr. Cem Kalyoncu
Lecturer	Asst. Prof. Dr. Cem Kalyoncu
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 5th semester
Type of teaching, expected class size	Face to face lectures, <70 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Labs: 2 hours per week 3. Self-Study: 3 hours per week 4. Examination Preparation time: 80 hours 5. Homework: 30 hours
Credit Points - ECTS	4 Credit Points – 8 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	Object Oriented Programming I
Catalogue Descriptions/Content	This course furthers the knowledge of participants in object oriented programming with emphasis on Modern C++ language following C++14 standard. Topics include: object oriented concepts; abstraction, inheritance, encapsulation, polymorphism, interfaces, const-correctness; object oriented programming in C++; classes, data members, constructors, member functions, overloading, virtual functions, pure virtual functions, abstract classes, overriding, shadowing, static members, use of const functions; operator overloading including array access, function call, type casting operators; multiple-inheritance, ADL, RTTI, templates, standard template library; basic containers, algorithms, memory management, threading
Course Learning Outcomes	Upon completion, the student should have gained: 1. Able to understand and build upon object oriented programming methodologies 2. Building C++ knowledge to create commercial software 3. Implementation of OOP concepts in C++ 4. Advanced OOP concepts such as multiple inheritance 5. Knowledge on C++ standard template library 6. Generic programming using templates
Study and examination requirements and forms of examination	• In class exercises • Two quizzes • Lab work • Midterm Examination • Homework • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture notes, homework, quiz
Reading List/ Recommended Text Book	The C++ Programming language, 4th Edition, Bjarne Stroustrup

Course Name	Analysis of Algorithms
Course Level	Undergraduate
Course Code	COMP 335
Semester	Fall
Person Responsible for the course	Assoc. Prof. Dr. Ezgi Deniz Ülker
Lecturer	Assoc. Prof. Dr. Ezgi Deniz Ülker
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 5th semester
Type of teaching, expected class size	Open Access, online <90 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 3 hours per week 3. Total Exercises and Examination Preparation time: 30 hours
Credit Points - ECTS	3 Credit Points – 4 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	Computer Programming
Catalogue Descriptions/Content	Understanding the basic concepts of recursion, recursive algorithms, iterative methods, understanding the concepts of iterative and recursive. Calculating the algorithmic complexity, Big (O) notations, worst case, average case and best case examples and comparative graphs of algorithmic complexity. Basic sorting algorithms; bubble sort, quick sort, selection sort, insertion sort with recursive and iterative comparisons. Calculating the algorithmic complexity of sorting methods, backtracking; 8x8 N-queens problem-solving, dynamic programming; Knapsack problem, Dijkstra's algorithm, longest common subsequence problem, matrix chain multiplication problem and expressing them using recursive method.
Course Learning Outcomes	On successful completion of this course, all students will have developed knowledge and understanding of: 1.Calculate the running time of an iterative algorithm. 2.Calculate the algorithmic complexity of an iterative algorithm. 3.Calculate the algorithmic complexity of a recursive algorithm. 4.Execute some examples by using recursive method. 5.Execute some examples of various types of the algorithms such as dynamic programming, greedy algorithms, backtracking.
Study and examination requirements and forms of examination	• In class exercises • Midterm Examination • Assignments • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture video links, lecture note sharing. MS Teams for recording the lectures.
Reading List/ Recommended Text Book	Main: An Introduction to analysis of a-Algorithms, Addison-Wesley Professional; 2nd edition, 2013, Robert Sedgewick, Philippe Flajolet. Supporting: Introduction to Algorithms - 3rd Edition; Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein; MIT Press, 2011.

Course Name	Database Management Systems
Course Level	Undergraduate
Course Code	COMP 337
Semester	Fall
Person Responsible for the course	Assist. Prof. Dr. Ferhun Yorgancıoğlu
Lecturer	Assist. Prof. Dr. Ferhun Yorgancıoğlu
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 5th semester
Type of teaching, expected class size	Face to face lectures, <65 Students
Workload	1.Lectures: 3 Lecture hours per week 2.Self-Study: 3 hours per week 3.Total Exercises and Examination Preparation time: 20 hours
Credit Points - ECTS	4 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	None
Catalogue Descriptions/Content	Foundations of a database system. Relational model: tables, tuples, attributes, domains, database schema, keys, schema diagrams, relational query languages, the relational algebra. Structured Query Language (SQL): overview, basic types and schema definitions, basic structure of SQL queries, queries on a single relation, queries on multiple relations, Cartesian product, the rename operation, string operations, ordering the display of tuples, set operations, null values, aggregate functions, aggregation with grouping, the having clause, nested subqueries, set membership, set comparison, test for empty relations, test for the absence of duplicate tuples, subqueries in the from clause, the with clause, scalar subqueries, modification of the database, insertions, deletions and updates, join expressions, views, transactions, integrity constraints, SQL data types and schemas, index definition in SQL, authorization, accessing SQL from a programming language. Database design using the Entity-Relationship (E-R) model: design process, the E-R model, entity sets, relationship sets, complex attributes, mapping cardinalities, participation constraints, E-R diagrams, reducing E-R diagrams to relational schemas, extended E-R features, design issues, alternative notations.
Course Learning Outcomes	Upon successful completion of the course, the student should have gained: 1.a general understanding of working with the relational databases 2. theoretical skills in writing queries using relational algebra 3. practical skills in writing queries for the relational databases using SQL 4. practice on how to draw an E-R diagram and how to reduce an E-R diagram to relational database schemas as a step towards database design 5. team-based project experience by developing a database application
Study and examination requirements and forms of examination	• Laboratory works • Database application program development • Midterm Examination • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture note sharing
Reading List/ Recommended Text Book	Main: A. Silberschatz, H.F. Korth, S. Sudarshan, Database Systems Concepts, 7th ed, McGraw-Hill, 2020. Supporting: J.A. Hoffer, R. Venkataraman, H. Topi, Modern Database Management, 13th ed, Pearson, 2020.

Course Name	Systems programming
Course Level	Undergraduate
Course Code	COMP339
Semester	Fall
Person Responsible for the course	Asst. Prof. Dr. Cem Kalyoncu
Lecturer	Asst. Prof. Dr. Cem Kalyoncu
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 5th semester
Type of teaching, expected class size	Face to face lectures, <70 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study:2 hours per week 3. Examination Preparation time: 40 hours 4. Homework: 12
Credit Points - ECTS	3 Credit Points – 4 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	COMP224
Catalogue Descriptions/Content	This course introduces system programming in the Linux environment with emphasis on C language. Topics include: program arguments, return codes, compiling programs on Linux. Understanding Linux environment and system calls. Basic terminal commands. POSIX processes, orphan and zombie processes, signals, signal handling. Low level IO calls, managing file systems using C and system calls. POSIX threads, mutexes; multithreading constraints and problems, synchronization, critical sections. Interprocess mechanisms; pipes, shared memory, UNIX sockets. Introduction to Internet communication protocols using TCP/IP and UDP/IP sockets. Creation and management of semaphores, using file tokens, the relation with shared memory.
Course Learning Outcomes	At the end of this course, the student should be able to implement low level programs that can interact with operating system and the other running applications. Upon completion, the student should have gained: 1. Familiarity with system programming terms, 2. Knowledge of system calls, 3. An ability to create and manage processes and threads, 4. An ability to perform inter-process communication, 5. The experience of using inter-process communication methods between computer.
Study and examination requirements and forms of examination	• In class exercises • Two quizzes • Midterm Examination • Homework • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture notes, homework, quiz
Reading List/ Recommended Text Book	• Linux : The Complete Beginner's Guide - The Black Book, Byron Francis, 2016 • Linux System Programming: Talking Directly to the Kernel and C Library, Robert Love, 2013 • The Linux Programming Interface, Micheal Kerrisk, 2010 • Unix Network Programming, by W. Richard Stevens , Bill Fenner, Andrew M. Rudoff, 2003

Course Name	Software Design and Architecture
Course Level	Undergraduate
Course Code	Seng305
Semester	Fall
Person Responsible for the course	Asst. Prof. Dr. Vesile Evrim
Lecturer	Asst. Prof. Dr. Vesile Evrim
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 5th semester
Type of teaching, expected class size	Face to face lectures, <70 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 3 hours per week 3. Total Exercises and Examination Preparation time: 35 hours
Credit Points - ECTS	3 Credit Points – 7 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	This course will provide a comprehensive introduction to software architecture and design, methods, processes, and notations. This course will explore design as the enumeration, evaluation, and selection of design alternatives to achieve quality attributes, including perspectives on design from performance improving, domain modelling, architectural styles, graphical user interface, and design patterns.
Course Learning Outcomes	At the end of this course student will be able to: 1. Familiarity with use cases and models, 2. Ability to design the dataflow in the software, 3. Knowledge on common architecture types, 4. Knowledge and ability to use design patterns. 5. Increase the communication skills among the other engineers
Study and examination requirements and forms of examination	• Midterm Examination • Project • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture note sharing
Reading List/ Recommended Text Book	Main: 3- Ian Sommerville, Software Engineering, 10th Edition, Addison Wesley, 2015. Supporting: 1- Barry Boehm, MBASE_Guidelines 2- https://sourcemaking.com/antipatterns

Course Name	Turkish
Course Level	Undergraduate
Course Code	COM106
Semester	Spring
Person Responsible for the course	Assoc. Prof. Dr. Osman Erciyas
Lecturer	Assoc. Prof. Dr. Osman Erciyas
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 6th semester
Type of teaching, expected class size	Online, >500 Students
Workload	1.Lectures: 3 Lecture hours per week 2.Self-Study: 2 hours per week 3.Total Exercises and Examination Preparation time: 38 hours
Credit Points - ECTS	2 Credit Points – 2 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	The course aims at providing the basic characteristics of written language and written communication and the differences between written language and spoken language. Expression: written and verbal expression; subjective expression; objective expression; paragraph; paragraph types (introduction, body, conclusion paragraphs). Defining texts and text types (informative texts, literary texts) conditions in texts (cohesion, consistency, intentionality, acceptability, contingency, information). Theoretic information about written expression, planned writing processes, and informative texts. Studying samples and writing applications, summarizing and preparing the plan, and correcting language and expression mistakes.
Course Learning Outcomes	The main aim of the course is to help non-Turkish speaking foreign students to gain basic reading, writing and speaking skills in Turkish language. The course is given in both Turkish and English languages in parallel so as to teach the vocabulary that students can use in everyday life through the comprehension of Turkish voice and sentence structure. On successful completion of this course, students will have developed knowledge and understanding of: LO1: Basic reading, writing and speaking skills in Turkish language, LO2: The vocabulary that can be used in everyday life through the comprehension of Turkish voice and sentence structure.
Study and examination requirements and forms of examination	• Midterm Examination • Final Examination
Media Employed	Projector, Moodle, Microsoft Teams
Reading List/ Recommended Text Book	Main: Supporting:

Course Name	Computer Networks
Course Level	Undergraduate
Course Code	COMP342
Semester	Fall
Person Responsible for the course	Assoc. Prof. Dr. Yönal Kırsal
Lecturer	Assoc. Prof. Dr. Yönal Kırsal
Language	English
Relation to Curriculum	The undergraduate degree program, Compulsory, 6th semester
Type of teaching, expected class size	Face-to-face lectures, <100 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 3 hours per week 3. Total Exercises and Examination Preparation time: 44 hours
Credit Points - ECTS	3 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	None
Catalogue Descriptions/Content	Physical Layer-the theoretical basis for data communication, guided transmission media, wireless transmission, Data Link Layer, data link layer design issues, MAC Sublayer, MAC Sublayer (Wireless), MAC Sublayer (Bridges), network layer design issues, Network Layer (Routing and Routers), Network Layer (Internetworking/IP), Transport Layer/Services/ Protocols. The Transport Service, elements of transport protocols, flow control and buffering, multiplexing, introduction to UDP, the TCP service model, the TCP protocol, performance issues, the application layer, the domain name system. Internetworking with IP (classes of IP addresses; IPV4 and IPV6), Cisco Packet Tracer Tutorial.
Course Learning Outcomes	On successful completion of the course, students should have gained: (1) An understanding of overarching frameworks for telecommunications network designs and operations, (2) An appreciation of the OSI framework by focusing on specific example implementations, (3) An understanding of various multi-service network topologies and how specific industrial network implementations fit within the broad topologies, (4) An accurate appreciation of how different switched networks are designed and implemented in order to provide internet services.
Study and examination requirements and forms of examination	• In-class exercises • Midterm Examination • Project • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture note sharing
Reading List/ Recommended Text Book	Main: 1. A. S. Tanenbaum, Computer Networks, 5th ed., Prentice Hall, 2011. 2.L. L. Peterson and B. S. Davie, Computer Networks: A Systems Approach, 5th ed.,Morgan Kaufmann, 2012.

Course Name	Principles of Programming Languages
Course Level	Undergraduate
Course Code	COMP364
Semester	Fall
Person Responsible for the course	Asst. Prof. Dr. Cem Kalyoncu
Lecturer	Asst. Prof. Dr. Cem Kalyoncu
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 6th semester
Type of teaching, expected class size	Face to face lectures, <70 Students
Workload	1. Lectures: 4 Lecture hours per week 2. Self-Study: 3 hours per week 3. Examination Preparation time: 45 hours 4. Homework: 20
Credit Points - ECTS	4 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	COMP124
Catalogue Descriptions/Content	The language evaluation, language design, evolution of programming languages, language categories. The names, variables, binding, scope, lifetime, referencing environments, constants, data types, enumeration types, array types, pointer and reference types. The arithmetic expressions, overloaded operators, relational and boolean expressions, type conversions. The selection statements, iterative statements. The subprograms, parameter passing methods, nested subprograms, blocks and scoping. The syntax, semantics, lexical and syntax analysis, syntax parsing, abstract syntax tree, recursive descent parser.
Course Learning Outcomes	Upon completion, the student should have gained: 1. Obtaining in depth knowledge about imperative language components 2. Learning about scoping rules, calling conventions, and parameter passing methods 3. Knowledge on functional programming languages 4. Ability to understand programming language grammar, semantics, and the methods to describe grammars 5. Ability to parse syntax of a programming language
Study and examination requirements and forms of examination	• In class exercises • Two quizzes • Midterm Examination • Homework • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture notes, homework, quiz
Reading List/ Recommended Text Book	• Concepts Of Programming Languages, Robert W. Sebesta, 2014, 10th edition, Pearson Education

Course Name	Software Quality Assurance and Testing
Course Level	Undergraduate
Course Code	SENG308
Semester	Spring
Person Responsible for the course	Assist. Prof. Dr. Zafer Erenel
Lecturer	Assist. Prof. Dr. Zafer Erenel
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 6th semester
Type of teaching, expected class size	Face to face lectures, <60 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 3 hours per week 3. Total Exercises and Examination Preparation time: 20 hours
Credit Points - ECTS	3 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	
Catalogue Descriptions/Content	Software errors, faults and failures. Software quality assurance, software quality requirements, software quality factors, components of the software quality assurance system. Pre-project components, Software project life cycle components, Infrastructure components, Management SQA components. The contract review process and its stages, Proposal draft reviews. Development plan and quality plan objectives, Software development risks , SQA components in the project life cycle, Integrating quality activities in the project life cycle. Software development methodologies. Verification, validation and qualification. Software testing strategies, Software test classifications, White box testing, Black box testing, Test case design, Automated testing, Alpha and beta site testing programs. Pre-maintenance software quality components, Maintenance software quality assurance tools. The quality of external participants' contributions, Risks and benefits of introducing external participants. Staff training and certification, Determining training and updating needs, Defining positions requiring certification, Delivery of training and certification programs
Course Learning Outcomes	On successful completion of this course, all students will have developed knowledge and understanding of: 1. an understanding of the software quality assurance activities. 2. awareness of different testing techniques. 3. ability to draw conclusions from software failures in different projects. 4. knowledge of pre-project stages.
Study and examination requirements and forms of examination	• In class examples • Midterm Examination • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture note sharing
Reading List/ Recommended Text Book	Main: Galin D.; Software Quality Assurance From theory to implementation; Pearson-Addison Wesley; 2003 Supporting: Naik K., Tripathy P.; Software Testing & Quality Assurance; Wiley; 2008

Course Name	Human Computer Interaction
Course Level	Undergraduate
Course Code	SENG 312
Semester	Spring
Person Responsible for the course	Assoc. Prof. Dr. Ezgi Deniz Ülker
Lecturer	Assoc. Prof. Dr. Ezgi Deniz Ülker
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 6th semester
Type of teaching, expected class size	Open Access, online <80 Students
Workload	1.Lectures: 3 Lecture hours per week 2.Self-Study: 6 hours per week 3.Total Exercises and Examination Preparation time: 50 hours
Credit Points - ECTS	3 Credit Points – 6 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	Overview of new technology in HCI in the context of past research. Input devices. Output devices. Reviewing and testing human computer interfaces. Guidelines and principles for good user interface design. Functionality. Design of systems using wireframes and storyboards. Implementation of systems, architecture, and navigation. Future of HCI. User-centered design for interaction, human computer interaction. Understanding the effects of human factors in developing and operating the information system. Practical issues encountered in man-machine interaction and user-interface design. Current trends in the development of interaction technologies.
Course Learning Outcomes	On successful completion of this course, all students will have developed knowledge and understanding of: 1.Understand the Computer and Human-Computer Interaction (HCI). 2.Understand the important aspects of implementation of human-computer interfaces. 3.Identify the various tools and techniques for interface analysis, design, and evaluation. 4.Identify the importance of working in teams and the role of each member within an interface development phase.
Study and examination requirements and forms of examination	• In class exercises • Midterm Examination • Assignments • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture video links, lecture note sharing. MS Teams for recording the lectures.
Reading List/ Recommended Text Book	Main:Steve Love, Understanding Mobile Human Computer Interaction Elsevier Butterworth HEINEMAN series. [ISBN 978-0-7506-6352-6] 2009 Supporting: Scott MacKenzie, Human Computer Interaction, An Empirical Research Perspective. Morgan Kaufman Publishers An imprint of Elsevier [ISBN 978]

Course Name	Java Programming
Course Level	Undergraduate
Course Code	Comp 471
Semester	Spring
Person Responsible for the course	Asst. Prof. Dr. Vesile Evrim
Lecturer	Asst. Prof. Dr. Vesile Evrim
Language	English
Relation to Curriculum	Undergraduate degree program, Elective, 6th semester
Type of teaching, expected class size	Face to face lectures, <70 Students
Workload	1.Lectures: 3 Lecture hours per week 2.Self-Study: 3 hours per week 3.Total Exercises and Examination Preparation time: 30 hours
Credit Points - ECTS	3 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	Introducing the imperative programming part of Java that is most similar to C language as well as the overall handling of memory in Java. Studying the object oriented features of Java: objects, classes, inheritance, method overriding, abstract methods and classes, interfaces. Introducing the inner classes: member classes, local classes, anonymous classes. Presenting exception handling and multithreaded programming in Java. Introducing technical features such as packages of classes and various useful predefined classes.
Course Learning Outcomes	At the end of this course student will be able to: 1.Have an ability to understand the fundamental concepts of java programming. 2.To understand how to design GUI components with the Java Swing API. 3.To learn how to implement object-oriented designs with Java. 4.To understand how to use Java for android applications 5.Have an ability to use polymorphism
Study and examination requirements and forms of examination	• Midterm Examination • Project • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture note sharing
Reading List/ Recommended Text Book	Main: 1-Harvey M. Deitel, "Java How to Program 7th edition 2- Linden. P, "Just Java 2", Sun M. Press,2004 Supporting Links: 1-https://www.w3schools.com/java/ 2-https://www.javatpoint.com/java-oops-concepts

Course Name	Engineering Economics
Course Level	Undergraduate
Course Code	ECON413
Semester	Fall
Person Responsible for the course	Dr. Kaan Kutlay
Lecturer	Dr. Kaan Kutlay
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 7th semester
Type of teaching, expected class size	Face to Face, <150 Students
Workload	1.Lectures: 3 Lecture hours per week 2.Self-Study: 4 hours per week 3.Total Exercises and Examination Preparation time: 40 hours
Credit Points - ECTS	3 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	Engineering economy principles. Cash-flow diagrams. Time effect on money. Formulas for reflecting time effect on money. How to value money that was spent before and how to value if it will be spent in the future while comparing different alternatives at present. Interest rate, simple interest rate, compound interest rate and compounding periods. How different compounding periods affecting the total amount of interest earned from the deposit. Why different alternatives need to be compared on economical basis. What is feasibility? Comparing different alternatives, examples. Minimum rate of return, attractive rate of return. Replacement and economic life concepts and problems about replacement concept by following different evaluation techniques The main purpose of this course is to introduce some of the basic concepts of economy for engineering students. Effect of time on money, methods for comparing different alternative, Benefit-Cost Analysis, Replacement and Depreciation concepts are discussed.
Course Learning Outcomes	At the end of this course, the student will be able to: 1.understand the some of the basic principles of economy, 2.understand and analyze the methods of comparing engineering projects' alternatives, 3.understand and analyze time effect on money by introducing interest rate, 4.evaluate how to make depreciation analysis 5.evaluate how to investigate replacement concept
Study and examination requirements and forms of examination	• Midterm Examination • Final Examination
Media Employed	Whiteboard, Projector, Moodle, Microsoft Teams
Reading List/ Recommended Text Book	Main: Leland T. Blank, Anthony Tarquin, Engineering Economy 8th Edition, Mc Graw Hill (2018) (Textbook) Supporting: William G. Sullivan, Elin M. Wicks, C. Patrick Koelling, Engineering Economy 16th Edition Pearson (2015) (Reference)

Course Name	Strategic Planning and Management
Course Level	Undergraduate
Course Code	BUSN461
Semester	Fall
Person Responsible for the course	Assoc. Prof. Dr. Gözde İnal Cavlan
Lecturer	Assoc. Prof. Dr. Gözde İnal Cavlan
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 7th semester
Type of teaching, expected class size	Face to Face, <150 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 4 hours per week 3. Total Exercises and Examination Preparation time: 56 hours
Credit Points - ECTS	3 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	Strategic Planning, setting goals, basic concepts of strategic management, developing a strategic plan, vision, mission, objectives, strategies and action plans, project management, types of project management, risk management, the six step process of risk management, risk management steps and tools, entrepreneurship, innovation, invention, the practice of innovation, entrepreneurial management, change management and leadership, roles and responsibilities for change, leadership style, strategic leadership, strategic leadership failure, global and social effects of engineering practices.
Course Learning Outcomes	On successful completion of this course, all students will have developed knowledge and understanding of: 1. To provide a basic understanding of the nature and dynamics of the strategy formulation and implementation processes as they occur in complex organizations. 2. To encourage students to think critically and strategically. 3. To develop the ability to identify strategic issues and design appropriate courses of action. 4. Demonstrate the ability to think critically in relation to a particular problem, situation or strategic decision through real-world scenarios. 5. Begin building a strategic plan for your area of responsibility. 6. To adapt change management.
Study and examination requirements and forms of examination	• Project • Midterm Examination • Final Examination
Media Employed	Whiteboard, Projector, Moodle, Microsoft Teams
Reading List/ Recommended Text Book	Main: Wheelen, T.L. and Hunger, D. J., (2012) Concepts in Strategic Management and Business Policy, Towards Global Sustainability, Boston: Pearson (Textbook) Supporting: David Hillson., "The Risk Management Handbook: A Practical Guide to Managing the Multiple Dimensions of Risk", Kogan Page, 1st edition, 2016 2. Melissa Schilling., "Strategic Management of Technological Innovation", 5th Edition, Kindle Edition, 2016

Course Name	Summer Training
Course Level	Undergraduate
Course Code	SENG360
Semester	Fall
Person Responsible for the course	Assoc. Prof. Dr. Ezgi Deniz Ülker
Lecturer	-
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 7th semester
Type of teaching, expected class size	Face to Face, <60 Students
Workload	1.Lectures: 0 Lecture hours per week 2.Self-Study: 40 hours per week 3.Total Exercises and Examination Preparation time: 2 hours
Credit Points - ECTS	0 Credit Points – 2 ECTS
Requirements according to the examination regulations	A student must complete 240 hours of training during summer term.
Pre-requisites	-
Catalogue Descriptions/Content	This is a 4 week practical experience course in the "real world" of computer or software engineering. For two ECTS credits, you must work at your internship site for 240 hours. In addition, you will need to submit your training report and do a short presentation on your work.
Course Learning Outcomes	On successful completion of this course, all students will have: 1 To develop software skills, 2 To discover the area of interest, 3 To tie theory to practice, 4 To seek career alternatives before graduation, 5 To increase individual level of awareness..
Study and examination requirements and forms of examination	• Project • Presentation
Media Employed	Whiteboard, Projector, Moodle, Microsoft Teams
Reading List/ Recommended Text Book	Main: Summer Training Log Book (to be filled by candidate and signed by the empoyer)

Course Name	Software Project Management
Course Level	Undergraduate
Course Code	SENG407
Semester	Fall
Person Responsible for the course	Asst. Prof. Dr. Vesile Evrim
Lecturer	Asst. Prof. Dr. Vesile Evrim
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 7th semester
Type of teaching, expected class size	Face to Face, <150 Students
Workload	1.Lectures: 3 Lecture hours per week 2.Self-Study: 3 hours per week 3.Total Exercises and Examination Preparation time: 65 hours
Credit Points - ECTS	3 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	It is intended students completing this course will be highly competent in all aspects of project management as defined in the syllabus, and be capable of taking full responsibility for medium to large-sized projects. Holders will be expected to be fully aware of, and competent to use, the current methodologies, techniques and tools required for effective project management.
Course Learning Outcomes	On successful completion of this course, all students will have developed knowledge and understanding of: 1. Understand organization structures 2. Understand software processes and effect on scheduling 3. Understand economic aspect of project management 4. Understand the important concepts of quality management 5- Improve the communication skills.
Study and examination requirements and forms of examination	• Project • Midterm Examination • Final Examination
Media Employed	Whiteboard, Projector, Moodle, Microsoft Teams
Reading List/ Recommended Text Book	Main: Kathy Schwalbe-Information Technology Project Management 8e-Cenage Learning (2018) (Textbook) Supporting:

Course Name	Graduation Project I
Course Level	Undergraduate
Course Code	SENG451
Semester	Fall
Person Responsible for the course	Assist. Prof. Dr. Vesile Evrim
Lecturer	All Faculty
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 7th semester
Type of teaching, expected class size	Face to Face, <30 Students
Workload	1.Lectures: 0 Lecture hours per week 2.Self-Study: 3 hours per week 3.Total Exercises and Examination Preparation time: 2 hours
Credit Points - ECTS	1 Credit Points – 3 ECTS
Requirements according to the examination regulations	A student must complete the first part of the graduation project successfully in order to do the presentation.
Pre-requisites	-
Catalogue Descriptions/Content	This is a 14 week project course mainly putting the foundations of the graduation project. The students are expected to prepare a project proposal, which includes basically the requirements of the software they will implement, the technologies they will use and the planning of the project development in practical experience course in the "real world" of computer or software engineering.
Course Learning Outcomes	On successful completion of this course, all students will have: 1 To develop software skills, 2 To discover the area of interest, 3 To tie theory to practice, 5 To develop skills to work on a project alone and in groups.
Study and examination requirements and forms of examination	• Project • Poster
Media Employed	Whiteboard, Projector, Moodle, Microsoft Teams
Reading List/ Recommended Text Book	Main: Graduation Project Handbook

Course Name	Artificial Intelligence
Course Level	Undergraduate
Course Code	Comp415
Semester	Fall
Person Responsible for the course	Asst. Prof. Dr. Vesile Evrim
Lecturer	Asst. Prof. Dr. Vesile Evrim
Language	English
Relation to Curriculum	Undergraduate degree program, Elective, 7th semester
Type of teaching, expected class size	Face to face lectures, <150 Students
Workload	1.Lectures: 3 Lecture hours per week 2.Self-Study: 3 hours per week 3.Total Exercises and Examination Preparation time: 30 hours
Credit Points - ECTS	3 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	Basic concepts of artificial intelligence; systems think/act rational, systems think/act human like, goal based, utility based, reflex agents. Environment types; static, discrete, accessible, episodic, and deterministic. Problem solving, problem, actions, goals, simple agent problem, multi agent problem. Uninformed search strategies; Breadth First Search, Depth First Search, Uniform Cost, Iterative Deepening. Informed search strategies; Greedy Search, A* search, hill climbing, annulated simulating, admissible heuristic, complexity, completeness, optimal algorithms, game playing, min max algorithm, alpha beta pruning algorithm. Propositional Logic, truth table, entailment, inference, valid, tautologies. First Order Logic, modus ponens, resolutions, conjunctive form, disjunctive form, horn form. Forward chaining, backward chaining, resolution by refutation, generalized modus ponens.
Course Learning Outcomes	At the end of this course student will be able to: 1.Gain an understanding of the key components of the artificial intelligence (AI). 2.Define a problem and environmental factors 3.Use the appropriate search method in achieving desired goals. 4.Represent knowledge using various techniques. 5.Learn inference methods to derive new sentences
Study and examination requirements and forms of examination	• Midterm Examination • Project • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture note sharing
Reading List/ Recommended Text Book	Main: S. Russell, P. Norvig, "Artificial Intelligence: A Modern Approach", Prentice-Hall, 2003.

Course Name	Narrowband Wireless Communications
Course Level	Undergraduate
Course Code	EE419
Semester	Fall
Person Responsible for the course	Assoc. Prof. Dr. Yönal Kırsal
Lecturer	Assoc. Prof. Dr. Yönal Kırsal
Language	English
Relation to Curriculum	The undergraduate degree program, Compulsory, 7th semester
Type of teaching, expected class size	Face-to-face lectures, <100 Students
Workload	1.Lectures: 3 Lecture hours per week 2.Self-Study: 3 hours per week 3.Total Exercises and Examination Preparation time: 40 hours
Credit Points - ECTS	3 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	None
Catalogue Descriptions/Content	This course introduces the characteristics and applications of transmission fundamentals, communication networks, the cellular concept and system design fundamentals frequency reuse, interference and system capacity. Protocols and the TCP/IP protocol suite. Antennas and radio propagation and large-scale path loss. Small-scale fading and multipath propagation. Doppler shift, mobile multipath channel parameters such as coherence bandwidth and coherence time. Diversity techniques and diversity combining. Spread spectrum communication techniques. Multiple access techniques TDMA, FDMA, CDMA, SDMA. Satellite Communications, Wireless LAN technologies.
Course Learning Outcomes	On successful completion of the course, students should be able to: (1) An ability to understand the basics of wireless communication, (2) An ability to acquire a good knowledge of wireless communication systems and applications, (3) An ability to understand the standards/technologies for various wireless computing systems, (4) An ability to be aware of trends in wireless computing systems and applications, (5) An ability able to compare the various access techniques and will learn the fundamentals of satellite communications.
Study and examination requirements and forms of examination	• In-class exercises • Midterm Examination • Quizzes • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture note sharing
Reading List/ Recommended Text Book	Main: Stallings, W., Data & Computer Communications, 10th Edition, Prentice Hall, 2013 Wireless Communications, by A. F. Molisch, 2005, John Wiley and Sons. ISBN-13 978-0-470-84887-6 (HB) or ISBN-13 978-0-470-84888-3 (PB).

Course Name	Engineering Ethics
Course Level	Undergraduate
Course Code	ENGG434
Semester	Spring
Person Responsible for the course	Assoc. Prof. Dr. Ertuğ Aydın
Lecturer	Assoc. Prof. Dr. Ertuğ Aydın
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 8th semester
Type of teaching, expected class size	Face to face lectures, <150 Students
Workload	1.Lectures: 3 Lecture hours per week 2.Self-Study: 4 hours per week 3.Total Exercises and Examination Preparation time: 38 hours
Credit Points - ECTS	3 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	Ethics and professionalism, moral reasoning, moral frameworks, ethical theories, the commitment to safety, risks, workplace responsibilities, honesty, equal opportunity: non-discrimination, confidentiality, and conflicts of interest, environmental ethics, green engineering, sustainable development, dilemma resolution, professional rights, whistleblowing. Code of ethics: The Institute of Electrical and Electronics Engineers, American Institute of Chemical Engineers, American Society of Civil Engineers, Software Engineering. Basic ethics training. Engineering professional training, job responsibilities and professionalism, labor law, and ethics. Case studies on the topics of engineering professional ethics, labor safety, and environmental protection. Computers and ethics, data protection, computer failures. Global issues.
Course Learning Outcomes	On successful completion of this course, all students will have developed knowledge and understanding of: 1.Develop comprehension of the professional and ethical responsibilities of engineers, including the code of 2.ethics of professional societies 3.Ability to address and resolve problems arising from questionable practice, and awareness of the need for lifelong learning. 4.Developing critical thinking skills and professional judgement understanding of practical difficulties, and awareness of the legal consequences of engineering solutions 5.Develop a professional ethical identity to carry forward in their working life
Study and examination requirements and forms of examination	• In-class case studies, group exercises , Midterm Examination 1 • Individual Project , Group Project • Presentation, Oral Examination • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture note sharing
Reading List/ Recommended Text Book	Main: 1.Mike W. Martin and Roland Schinzinger., "Ethics in Engineering", fourth edition, 2005 2.Barbara MacKinnon and Andrew Fiala, "Ethics: Theory and Contemporary Issues", Cengage Learning; 9th edition, 2017 3.Charles E. Harris Jr., Michael S. Pritchard., Michael J. Rabins., "Engineering Ethics: Concepts and Cases", 4th Edition, Wadsworth Publishing, 2008 Supporting: 1.Bowen, W. Richard, "Engineering Ethics", 1st Edition, Springer International Publishing, 2014 2.https://www.nspe.org/resources/ethics (NSPE, National Society of Professional Engineers)

Course Name	Graduation Project II
Course Level	Undergraduate
Course Code	SENG450
Semester	Spring
Person Responsible for the course	Assist. Prof. Dr. Cem Kalyoncu
Lecturer	All Faculty
Language	English
Relation to Curriculum	Undergraduate degree program, Compulsory, 7th semester
Type of teaching, expected class size	Face to Face, <30 Students
Workload	1.Lectures: 0 Lecture hours per week 2.Self-Study: 6 hours per week 3.Total Exercises and Examination Preparation time: 10 hours
Credit Points - ECTS	3 Credit Points – 10 ECTS
Requirements according to the examination regulations	A student must complete the second part of the graduation project successfully in order to do the presentation.
Pre-requisites	-
Catalogue Descriptions/Content	This is a 14 week project course mainly putting the foundations of the graduation project. The students are expected to write a project report, and show a power-point (or relevant) presentation with the actual demonstration of the software that they will develop. In this demonstration, the faculty members ask various questions to evaluate the knowledge of the student.
Course Learning Outcomes	On successful completion of this course, all students will have: 1 To develop software skills, 2 To discover the area of interest, 3 To tie theory to practice, 5 To develop skills to work on a project alone and in groups.
Study and examination requirements and forms of examination	• Project • Presentation
Media Employed	Whiteboard, Projector, Moodle, Microsoft Teams
Reading List/ Recommended Text Book	Main: Graduation Project Handbook

Course Name	Internet Programming
Course Level	Undergraduate
Course Code	Comp464
Semester	Spring
Person Responsible for the course	Asst. Prof. Dr. Vesile Evrim
Lecturer	Asst. Prof. Dr. Vesile Evrim
Language	English
Relation to Curriculum	Undergraduate degree program, Elective, 8th semester
Type of teaching, expected class size	Face to face lectures, <75 Students
Workload	1.Lectures: 3 Lecture hours per week 2.Self-Study: 3 hours per week 3.Total Exercises and Examination Preparation time: 30 hours
Credit Points - ECTS	3 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	-
Catalogue Descriptions/Content	This course is an introduction to the core technologies, tools, techniques and languages needed for the design and implementation of static and dynamic Web pages, as well as Web applications. Hypertext Markup Language (HTML), Cascading Style Sheets (CSS), Extensible Markup Language (XML), Extensible Stylesheet Language transformations (XSLT), JavaScript and AJAX are covered for programming on the client side. The list of technologies covered may change as new technologies become available and current ones become obsolete. Web servers, XML Web services, a scripting language and a corresponding Web application framework are covered for programming on the server side. Issues that are dealt with concerning server side programming also include session tracking, authentication, authorization, and database connectivity.
Course Learning Outcomes	At the end of this course student will be able to: 1. An ability to understand the fundamental concepts of client-side web programming. 2. An ability to understand the fundamental concepts of server-side web programming. 3. An ability to get instant information changes from sites through an API 4.To be able to integrate client and server side programming
Study and examination requirements and forms of examination	• Midterm Examination • Project • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture note sharing
Reading List/ Recommended Text Book	Main:Porter Scobey, Pawan Lingras. Web Programming and Internet Technologies: An E-commerce Approach, September 2016 Supporting Links: w3schools.comn/php

Course Name	Artificial Neural Networks
Course Level	Undergraduate
Course Code	COMP448
Semester	Spring
Person Responsible for the course	Assist. Prof. Dr. Zafer Erenel
Lecturer	Assist. Prof. Dr. Zafer Erenel
Language	English
Relation to Curriculum	Undergraduate degree program, Technical Elective, 8th semester
Type of teaching, expected class size	Face to face lectures, <120 Students
Workload	1.Lectures: 3 Lecture hours per week 2.Self-Study: 3 hours per week 3.Total Exercises and Examination Preparation time: 16 hours
Credit Points - ECTS	3 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	Linear Algebra
Catalogue Descriptions/Content	Background on Linear Algebra.History of Neural Networks. Articial and biological neural networks. Biological Neurons and Artificial Neurons. Models of single neurons. Different neural network models. Mc-Culloch-Pitts neuron, Hebb NET, Single Layer Perceptrons, Adaline, Least mean square algorithm. Learning rates. Activation Functions, Logic Gates, Threshold, Bias, Limitation of Linear models, The XOR problem. Multilayer Perceptrons, Back-propagation algorithm. Binary Classification, Multi-Class Classification. Unsupervised Learning strategies, Kohonen Self-Organising Maps, The SOM algorithm, Learning vector quantization, Hamming NET and Probabilistic Neural Networks.
Course Learning Outcomes	On successful completion of this course, all students will have developed knowledge and understanding of: (1) An ability to distinguish the main differences between artificial neural networks and biological neural networks and to understand the areas of neural nets, (2) An ability to create Mc-Culloch-Pitts, Hebb NET, Perceptron and Adaline architectures, (3) An aptitude for implementing Kohonen Self-Organizing Maps and its variations, (4) An ability to implement Learning Vector Quantization and Probabilistic Neural Networks, (5) An understanding of Backpropagation Neural Net with its variations, (6) Demonstration of competence in programming Single Layer and Multi Layer Neural Nets.
Study and examination requirements and forms of examination	• In class exercises • Midterm Examination • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture note sharing
Reading List/ Recommended Text Book	Main: Cole M.R.; Hands-On Neural Network Programming with C#, Packt Publishing, 2018 Supporting: Fausett, L..; Fundamentals of Neural Networks, Prentice Hall, 1994

Course Name	Principles of Digital Image Processing
Course Level	Undergraduate
Course Code	EE431
Semester	Fall
Person Responsible for the course	Asst. Prof. Dr. Cem Kalyoncu
Lecturer	Asst. Prof. Dr. Cem Kalyoncu
Language	English
Relation to Curriculum	Undergraduate degree program, Elective, 8th semester
Type of teaching, expected class size	Face to face lectures, <50 Students
Workload	1. Lectures: 3 Lecture hours per week 2. Self-Study: 3 hours per week 3. Examination (quiz, midterm, final) preparation time: 40 hours 4. Homework: 20 hours
Credit Points - ECTS	3 Credit Points – 5 ECTS
Requirements according to the examination regulations	A student must have attended at least 70% of the lectures to sit in the exams.
Pre-requisites	
Catalogue Descriptions/Content	This course introduces the principles of digital image processing applications and their implementations mainly in C++. Topic covers: Image sampling and quantization; interpolation techniques, nearest neighbour interpolation, bilinear interpolation; Histograms, understanding image histogram, contrast stretching, brightness and contrast, gamma, histogram equalization. Filtering in spatial domain, low pass filter, high pass filter, band pass filter, box filter, edge detection techniques. Color theory, human color vision, digital image color systems: RGB, HSI, HSV, CMYK. Image morphology, thresholding, erosion, dilation, opening and closing operations, and/or/not operations. Information theory, Shannon'e entropy, Huffmann compression, compression techniques, lossy/lossless compression.
Course Learning Outcomes	On successful completion of this course, all students will have developed knowledge and understanding of: 1. Familiarity with image processing terms 2. Knowledge of histograms and histogram processing 3. Ability to understand and perform filters on grayscale and binary images 4. Ability to understand and implement image processing algorithms 5. Having theoretical understanding of topics related to image processing, such as color theory, information theory
Study and examination requirements and forms of examination	• In class exercises • Two quizzes • Midterm Examination • Homework • Final Examination
Media Employed	Whiteboard, Projector and Moodle for Lecture notes, homework, quiz and project
Reading List/ Recommended Text Book	Main: R. C. Gonzalez and R. E. Woods, Digital Image Processing, 3rd Edition, Prentice Hall, 2008

DEPARTMENT OF
SOFTWARE ENGINEERING

ACADEMIC STAFF DETAILS

Assoc. Prof. Dr. Ezgi Deniz Ülker		
BSc Degree	Girne American University, North Cyprus	2008
MSc Degree	Girne American University, North Cyprus	2010
PhD Degree	Girne American University, North Cyprus	2013
Department	Software Engineering	
Research Area	Artificial Intelligence, Optimization, Bioinformatics	

Prof. Dr Hüseyin Ademgil		
BSc Degree	Eastern Mediterranean University, North Cyprus	2005
MSc Degree	University of Kent, UK	2006
PhD Degree	University of Kent, UK	2010
Department	Software Engineering	
Research Area	Optical devices, Optical Fibers, Photonics, Optical based Sensors	

Prof. Dr. Sadık Ülker		
BSc Degree	University of Virginia, USA	1996
MEE Degree	University of Virginia, USA	1999
PhD Degree	University of Virginia, USA	2002
Department	Software Engineering	
Research Area	Electromagnetic Field Theory, Microwaves, Antennas and Propagation, Artificial Intelligence	

Prof. Dr Akın Cellatoğlu		
BSc Degree	Eastern Mediterranean University, North Cyprus	1996
MSc Degree	University of Surrey, UK	1998
PhD Degree	University of Surrey, UK	2003
Department	Software Engineering	
Research Area	Multimedia Communications, wireless networks, 3D Display Panel	

Assist. Prof. Dr. Ferhun Yorgancıoğlu		
BS Degree	Eastern Mediterranean University, North Cyprus	2000
MS Degree	Eastern Mediterranean University, North Cyprus	2002
PhD Degree	Eastern Mediterranean University, North Cyprus	2008
Department	Software Engineering	
Research Area	Computer Network systems	

Asst. Prof. Dr. Cem Kalyoncu		
BSc Degree	Cyprus International University, North Cyprus	2005
MSc Degree	Cyprus International University, North Cyprus	2010
PhD Degree	Eastern Mediterranean University, North Cyprus	2015
Department	Computer Engineering	
Research Area	Machine learning, image processing, computer graphics	

Asst. Prof. Dr. Zafer Erenel		
BSc Degree	İstanbul University, Turkey	1999
MSc Degree	Isik University, Turkey	2005
PhD Degree	Eastern Mediterranean University, North Cyprus	2010
Department	Computer Engineering	
Research Area	information processing, text categorization and machine learning	

Asst. Prof. Dr. Eren Küren		
BSc Degree	European University of Lefke, North Cyprus	2003
MSc Degree	European University of Lefke, North Cyprus	2005
PhD Degree	European University of Lefke, North Cyprus	2019
Department	Computer Engineering	
Research Area	information processing, text categorization and machine learning	