



LEFKE AVRUPA ÜNİVERSİTESİ
EUROPEAN UNIVERSITY OF LEFKE

**DEPARTMENT OF ELECTRONICS &
SOFTWARE ENGINEERING**

PROGRAM INFORMATION

www.eul.edu.tr

PROGRAM INFORMATION

Program Name and Degree Awarded

Software Engineering – Bachelor of Science

Duration of Studies

4 years – 8 semesters

Total Credits / ECTS

144/240

Language of Instruction

English

Mission and Vision

Mission

The mission of the Software Engineering Programme is to cultivate graduates who are innovative problem solvers equipped with strong mathematical, analytical, and programming foundations, complemented by effective communication, critical thinking, leadership, and decision-making skills. Our graduates will be prepared to design, develop, and manage high-quality software systems, while upholding professional ethics and demonstrating social responsibility in an increasingly digital and interconnected world.

Vision

The vision of the Software Engineering Programme is to be recognized as a leading centre of excellence in the Eastern Mediterranean region for software engineering education and research. We aspire to produce globally competitive graduates who remain at the forefront of technological advances, committed to lifelong learning, professional integrity, and contributing to the sustainable development of society through innovative and ethical software solutions.

Program Objectives

The Software Engineering Programme aims to provide students with a strong foundation in the principles and practices of the profession. You will learn how modern software systems are designed, built, and maintained, with a focus on the entire software lifecycle — from requirements and analysis to design, implementation, testing, and management. The curriculum blends engineering principles, programming, software construction, and project leadership, while also highlighting the importance of teamwork and collaboration. Our goal is to give you a balanced education that combines both theory and practice.

The programme is internationally accredited by **ASIIN** and carries the **EURO-INF label**, ensuring its quality and recognition. To keep it relevant, the curriculum is continuously updated based on feedback from students, alumni, industry experts, and academic staff. Regular reviews and improvements guarantee that the courses reflect the latest research areas, technological developments, and real-world needs.

In line with the mission and vision of the department, the programme's educational objectives can be summarized as follows:

- To be educate 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.
- 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.
- 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.
- To work towards demonstrating leadership and entrepreneurship in their profession

These objectives ensure that graduates leave the programme with the technical knowledge, analytical skills, and professional values needed to adapt, grow, and lead in a fast-changing digital world.

Program Learning Outcomes

The program outcomes for the ECE department are listed below. 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 st SEMESTER				2 nd SEMESTER			
Course Code	Course Name	CREDIT	ECTS	Course Code	Course Name	CREDIT	ECTS
COMN191	ACADEMIC ENGLISH I	(3,0)3	3	UHTC01	HISTORY	(2,0)2	2
COMN111	CHEMISTRY	(3,0)3	4	COMN192	ACADEMIC ENGLISH II	(3,0)3	3
ENGG131	PHYSICS I	(3,0)3	4	COMN122	PHYSICS II	(3,0)3	5
ENGG117	COMPUTING FOUNDATIONS	(3,2)4	6	COMP124	COMPUTER PROGRAMMING	(3,2)4	6
SENG119	INTRODUCTION TO PROFESSION	(2,0)0	2	ENGG122	PHYSICS II LAB	(0,2)1	2
ENGG111	CHEMISTRY LAB	(0,2)1	2	MATH104	LINEAR ALGEBRA	(3,0)3	5
ENGG121	PHYSICS I LAB	(0,2)1	2	MATH110	CALCULUS II	(3,2)4	7
MATH101	CALCULUS I	(3,2)4	7				
TOTAL		19	30	TOTAL		20	30
3 rd SEMESTER				4 th SEMESTER			
SENG205	DIGITAL LOGIC DESIGN	(3,2)4	6	SENG214	OPERATING SYSTEMS	(3,3)3	5
SENG209	DISCRETE MATHEMATICS	(3,0)3	6	SENG218	OBJECT ORIENTED PROGRAMMING I	(3,2)4	6
SENG217	DATA STRUCTURES	(3,2)4	7	UFRC02	UNIVERSITY ELECTIVE II	(3,0)3	4
UFRC01	UNIVERSITY ELECTIVE I	(3,0)3	4	MATH224	ENGINEERING MATHS	(3,0)3	5
MATH201	ORDINARY DIFFERENTIAL EQUATIONS	(3,2)4	5	MATH226	PROBABILITY & STATISTIC METHODS	(3,0)3	5
UHTC02	TURKISH	(2,0)2	2	SENG212	SOFTWARE REQUIREMENTS ANALYSIS AND SPECIFICATION	(3,0)3	5
TOTAL		20	30	TOTAL		19	30
5 th SEMESTER				6 th SEMESTER			

SENG315	APPLIED OBJECT ORIENTED PROGRAMMING	(2, 2) 3	7	SENG342	COMPUTER NETWORKS	(3, 0) 3	6
SENG335	ANALYSIS OF ALGORITHMS	(3, 0) 3	5	SENG364	PRINCIPLES OF PROGRAMMING LANGUAGES	(3, 1) 4	7
SENG337	DATABASE MANAGEMENT SYSTEMS	(3, 2) 4	6	SENG308	SOFTWARE QUALITY ASSURANCE AND TESTING	(3, 0) 3	6
SENG339	SYSTEMS PROGRAMMING	(3, 0) 3	6	SENG312	HUMAN COMPUTER INTERACTION	(3, 0) 3	6
SENG305	SOFTWARE DESIGN AND ARCHITECTURE	(3, 0) 3	6	SWEL01	TECHNICAL ELECTIVE I	(3, 0) 3	5
TOTAL		16	30			16	30
7th SEMESTER				8th SEMESTER			
ENGG413	ENGINEERING ECONOMICS	(3, 0) 3	5	ENGG434	ENGINEERING ETHICS	(3, 0) 3	5
SENG451	GRADUATION PROJECT I	(0, 3) 1	4	SENG450	GRADUATION PROJECT II	(0, 6) 3	5
SENG360	SUMMER TRAINING	(0, 1) 0	1	SWEL05	TECHNICAL ELECTIVE V	(3, 0) 3	5
SENG407	SOFTWARE PROJECT MANAGEMENT	(3, 0) 3	5	SWEL06	TECHNICAL ELECTIVE VI	(3, 0) 3	5
SWEL02	TECHNICAL ELECTIVE II	(3, 0) 3	5	SWEL07	TECHNICAL ELECTIVE VII	(3, 0) 3	5
SWEL03	TECHNICAL ELECTIVE III	(3, 0) 3	5	SWEL08	TECHNICAL ELECTIVE VIII	(3, 0) 3	5
SWEL04	TECHNICAL ELECTIVE IV	(3, 0) 3	5				
TOTAL		16	30	TOTAL		18	30

Laboratory and Equipment Capacity

The Department of Software Engineering offers modern, fully equipped laboratories that give you hands-on experience with the latest technologies. We have four Windows labs (each with space for 30 students) and a dedicated Macintosh lab (also 30 seats), ensuring you'll always have access to the right tools for your courses and projects.

Lab sessions are guided by your instructors or teaching assistants, depending on the course, so you'll always have expert support. Our equipment is carefully chosen to match course and project needs, giving you the chance to work on real, practical challenges.

Beyond computer labs, you'll also find cutting-edge facilities like 3D Printers (X30 and X23), a Laser Cutting Machine, a Bench Lathe Machine, CNC Router for metal and plastic cutting, Benchtop Drill Presses, and professional soldering stations. These resources open the door for creativity, innovation, and building real prototypes — bringing your software engineering ideas to life.

Career Opportunities

A degree in Software Engineering opens the door to a wide range of exciting career paths. As a graduate, you can work as a software developer, web or mobile app designer, systems analyst, or database administrator. If you enjoy problem-solving and building new technologies, you'll find opportunities in industries such as finance, healthcare, gaming, telecommunications, and even

aerospace. Since almost every modern business relies on software, your skills will always be in high demand.

Beyond traditional roles, software engineers are also at the heart of emerging fields like artificial intelligence, cybersecurity, cloud computing, and data science. You can choose to work for global tech companies, innovative startups, or even launch your own business. With strong foundations in both theory and practice, you'll be prepared for a flexible and rewarding career, with endless chances to grow, specialize, and make an impact on the world.

Contact Information

Assist. Prof. Dr. Vesile Evrim

Address: European University of Lefke, Lefke, Northern Cyprus TR-10 Mersin, Turkey

E-mail: vevrim@eul.edu.tr

Tel: +90 392 660 2000-2507

Fax: +90 392 660 2503

COURSE CATALOGUE DESCRIPTIONS

COMN191 ACADEMIC ENGLISH I

This course is intended for academically oriented students and it aims to bridge the gap between general and academic English. The course aims at developing the skills required for academic study, including note-taking, essay writing, as well as teaching strategies for undertaking research and dealing with unfamiliar academic vocabulary. The course also aims at teaching the features of guided writing, reading strategies such as predicting, skimming, and scanning. At the end of this course the students are expected to be able to; develop strategies, to improve the ability to comprehend complex academic texts, to develop strategies to produce more coherent writing and, make clear, appropriate, relevant notes from academic texts, and to adopt various approaches to deal with new or unknown vocabulary by practising effective use of dictionaries, and through making effective vocabulary records.

COMN111 CHEMISTRY

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.

ENGG117 COMPUTING FOUNDATIONS

Introduction to general problem-solving concepts, algorithms and its applications. Computer terminology, units, and number systems. Steps in problem-solving. Problem solution, pseudocode, algorithms, flowcharts, data types, and control structures. History of computers and programming. A simple C program layout, syntax and rules. C language basics, native types, identifiers, declarations, variables, expressions, and assignments. Basic console input/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, break/continue statements. Tracing C code.

SENG 119 INTRODUCTION TO PROFESSION

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.

ENGG111 CHEMISTRY LAB

By the end of this course, students will have a general understanding of experimental methods. Students will have the basic knowledge on concepts such as chemistry, quantitative techniques, distillation, density, chromatography, aqueous solutions, volumetric methods, acid base equilibrium and titrations.

ENGG121 PHYSICS I LAB

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.

ENGG131 PHYSICS I

This course aims to introduce the fundamental concepts of physics necessary for engineering science and to provide essential background for engineering students. The course provides deep understanding about kinematics and dynamics of one dimensional, two dimensional, circular and rotational motion. Also, the course aims to show the students the engineering applications of the course material.

MATH101 CALCULUS I

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.

UHTC01 HISTORY

The course provides a detailed exposure on the history of the construction of the Turkish Republic under the light of Kemal Atatürk's principles. The aim of the course is to introduce a brief history of Turkish Republic and Cyprus. Social, economic and political aspects and effects of Western Civilization on Turkey and Cyprus. Relations with Middle East.

COMN192 ACADEMIC ENGLISH II

This course is the continuation of the COM101 English I course. Similar issues are focused on as in the former course with a higher tone of language. This course integrates all four language skills and teaches students how to integrate skills and content in real-world academic contexts. High-interest and intellectually stimulating authentic materials are used to familiarize students with academic content. The course also aims at developing the ability to participate in exchanges of information and opinions in the context of the specific field, and to write instructions, descriptions and explanations about topics in the related field. Extra importance is put on teaching students terminology related to the specific field.

COMN122 PHYSICS II

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.

COMP124 COMPUTER PROGRAMMING

Structured programming using C. Expressions: constants, data types, type modifiers, const qualifier, operators, order of evaluations, type conversions, type casts. Statements: selection, iteration, jump, label, expression, block. Selective statements: if-else, switch-case, conditional operator, nested forms. Iterative statements: for, while, do-while, infinite loops, comma operator, break, continue, nested forms. Functions: definitions, formal parameters, actual arguments, call-

by-value, function prototypes. Scope rules, storage classes. Arrays: declarations, initialization lists, define directive, arrays as function arguments, call-by-reference. Strings: character arrays, null character, string-handling functions. Pointers: pointer variable declarations, array/pointer relationship, pointer arithmetic, array-subscript and pointer-offset notations, pointers as function arguments. Structures: user-defined types, typedef, structures as function arguments, array of structures, pointer to a structure. Unions, bit-fields, enumerations, preprocessing directives, and standard header files.

ENGG122 PHYSICS II LAB

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.

MATH104 LINEAR ALGEBRA

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.

MATH110 CALCULUS II

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.

SENG205 DIGITAL LOGIC DESIGN

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.

SENG209 DISCRETE MATHEMATICS

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 and relations.

SENGP217 DATA STRUCTURES

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.

MATH201 ORDINARY DIFFERENTIAL EQUATIONS

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.

UHTC02 TURKISH

To show the characteristics and rules of operation of Turkish language with examples; to give the students the ability and habit to express their feelings and thoughts accurately and effectively; developing vocabulary through written and oral texts; The aim of this course is to teach the rules of reading texts or the programs they listen to correctly. COM 106 course aims to provide basic Turkish reading, speaking and writing skills for international students.

SENG214 OPERATING SYSTEMS

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.

SENG218 OBJECT ORIENTED PROGRAMMING I

Object-oriented programming using C++. C++ console input/output streams. C++ functions: inlining, overloading, default arguments, reference parameters. C++ operators new/delete. C++ vectors. C++ strings. Classes and objects: data abstraction, data members and member functions, encapsulation, hiding data, member access methods: public, private, and protected, accessor/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: static vs dynamic binding, pure virtual functions, abstract classes, virtual destructor.

MATH224 ENGINEERING MATHS

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.

MATH226 PROBABILITY AND STATISTIC METHODS

Introduction to statistic and data analysis, Measures of location, Sample mean, Sample median, Mode value, Probability and sample space, Probability of an event, Additive rules, Bayes rule, Random variables and probability distributions, Discrete probability distributions, Continuous probability distributions, Mean, variance and covariance of random variables, Discrete uniform distributions, Binomial distributions, Hypergeometric distributions, Poisson distributions, Normal distribution, Areas under the normal curve, Applications of normal distributions, Gamma and exponential distributions, Chi-squared and lognormal distributions.

SENG212 SOFTWARE REQUIREMENTS ANALYSIS AND SPECIFICATION

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, feasibility study.

SENG315 APPLIED OBJECT ORIENTED PROGRAMMING

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

SENG335 ANALYSIS OF ALGORITHMS

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.

SENG337 DATABASE MANAGEMENT SYSTEMS

Foundations of a database-management system: view of data, data models, data abstraction, instances and schemas, data-definition and data-manipulation languages. Relational model: tables, tuples, attributes, primary and foreign keys, schema diagrams, relational query languages, relational algebra. Structured Query Language (SQL): built-in types, schema definitions, structure of SQL queries, queries on a single/multiple relation(s), rename operation, string operations, ordering tuples, attribute specification, where-clause predicates, set operations, null values, aggregate functions, aggregation with grouping, having clause, nested subqueries, set membership, set comparison, test for empty relations, test for the absence of duplicates, subqueries in the from clause, with clause, scalar subqueries, modification of the database, join expressions, views, integrity constraints. Entity-Relationship (E-R) model: design process, entity sets, relationship sets, complex attributes, mapping cardinalities, participation constraints, E-R diagrams, reducing E-R diagrams to relational schemas, extended E-R features, E-R design issues, alternative notations.

SENG 339 SYSTEMS PROGRAMMING

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.

SENG305 SOFTWARE DESIGN AND ARCHITECTURE

Understanding the importance of software design, tools that are used. Using use cases and scenarios, use case diagrams, actors, extending and including use cases, marking system boundaries. System model; context diagram, multi-level Data flow diagram, sequence diagram, class diagram. Entity-relationship model diagrams, Software architecture models; client-server, layered, blackboard, pipe and filter, implicit invocation, batch and sequence. Design Pattern (creative, behavioral, structural): Singleton, façade, bridge, observer, mediator. Graphical User Interface, design details, error messages.

SENG342 COMPUTER NETWORKS

An understanding of foundations and applications of computer networks, classical network topologies, network performances and network layer services. The reference OSI and TCP/IP models. Ethernet: physical properties and multiple access. Wireless topologies; Bluetooth, Wi-Fi, Wi-MAX and cellular networks. Application of Computer Networks, Types of Networks : PAN, LAN, MAN, WAN, Internet. Switching concepts, bridges and LAN switches. Coding, framing, error detection and fundamental data security. Internetworking with IP (classes of IP addresses; IPV4 and IPV6), Cisco Packet Tracer Tutorial.

SENG 364 PRINCIPLES OF PROGRAMMING LANGUAGES

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.

SENG308 SOFTWARE QUALITY ASSURANCE AND TESTING

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

SENG312 HUMAN COMPUTER INTERACTION

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.

ECON413 ENGINEERING ECONOMICS

Basic concepts of economy for engineering students. Interest rate, simple and compound interests are discussed. This course also introduces students to the key concept and topic of microeconomics such as opportunity cost, production possibility frontier, demand and supply, elasticity, utility, preferences, production, costs.

SENG407 SOFTWARE PROJECT MANAGEMENT

Components and challenges of software management, dealing with people involved in project development, time management. Project management knowledge areas, project management process groups, organizational structures, Software development life cycle, standard models; waterfall, incremental, prototyping, spiral, agile, scrum, Proactive and reactive project management. Risk management and analysis, proactive and reactive risk management strategies, methods to identify and quantify risks, risk mitigation techniques, balancing risk management

overhead. Planning, Project scheduling, project size estimation, Gantt charts, network diagrams, cost analysis, effort estimation, COCOMO, Documenting software projects, managing multiple teams in scheduling. Software management metrics; lines of code, function points, capability maturity model.

ENGG434 ENGINEERING ETHICS

Ethics and professionalism, moral reasoning, moral frameworks, ethical theories, commitment of 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, environmental protection. Computers and ethics, data protection, computer failures. Global issues.

SENG451 GRADUATION PROJECT I

It provides students with the opportunity and time to take the first steps of their design work through literature research.

SENG450 GRADUATION PROJECT II

Through project developed in a specific technology, students gain the ability to access knowledge, engage in continuous learning, and enhance their personal and professional development.

SENG360 SUMMER TRAINING

The subjects that students have learned in the courses are put into practice through hands-on applications at the companies. The objective of this training is to enhance students' practical skills and provide them with valuable exposure to real-world

SOME OF THE ELECTIVE COURSES

EE431 DIGITAL IMAGE PROCESSING

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 neighbor 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's entropy, Huffman compression, compression techniques, lossy/lossless compression.

COMP415 ARTIFICIAL INTELLIGENCE

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 simmuling, 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.

COMP448 ARTIFICIAL NEURAL NETWORKS

Background on Linear Algebra. History of Neural Networks. Artificial 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.

EE419 WIRELESS COMMUNICATION

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.

COMP464 INTERNET PROGRAMMING

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.

ECE408 DIGITAL SIGNAL PROCESSING

Discrete-time signals, the relationship between continuous and discrete-time transforms, and the sampling theorem. System properties. The impulse response of discrete time, linear time-invariant (LTI) systems and the convolution sum. The z-transform and its properties. Design and analysis of LTI systems using the ztransform. Filter structures. Design of FIR and IIR digital filters.

Characterization using linear difference equations. Signal analysis using the discrete Fourier transform and the fast Fourier transform (FFT). Analysis and simulation using the MATLAB software package.

COMP461 AUTOMATA THEORY

Sets, relations, functions, graphs, trees, strings, languages. Parse trees. Deterministic Finite state automata (DFA), non-deterministic finite state automata, equivalence of deterministic finite state automata and non-deterministic finite state automata, minimization of finite automata. Mealy and Moore machines. Regular expressions and regular grammars. Conversion of DFA to regular expression, conversion of regular expression to finite automata. Closure property of regular languages. Context-free grammars (CFG) and languages. ϵ -productions, derivations and sentential forms. Normal forms of CFG, Chomsky Normal Form, Greibach Normal Form. Pushdown automata, Deterministic Pushdown automata, Non-Deterministic Pushdown Automata, Construction of PDA from CFG. Construction of CFG from PDA.

COMP440 DATABASE SECURITY

Entity relationship model, SQL, query, nested queries, select, from, where, group by, having, order, like, set operation, referential integrity, cascade, primary key, candidate key, super key, foreign key, constraints. Discretionary access control; grant, revoke, privilege, create user roles,

permission, inference, prepared statement. Mandatory Access Control, security levels, top secret, secret, classified, unclassified, polyinstantiation, SQL injection, covert channels, Trojan horses, star model. Backup and recovery methods, auditing, inspection, concurrency control, isolation, durability, commit, rollback, transaction log, system failures, check point, power failure, undo, redo, transaction recovery, lost update, uncommitted data, inconsistent analysis.

COMP469 DISCRETE EVENT SYSTEM SIMULATION

The theories of system simulation. The simulation examples, general principals, simulation models, applications, system components, discrete systems, continuous systems. The queuing systems, single channel queues, two channel queues, event scheduling/time-advance algorithm, list processing, data structures. The statistical models in simulation, random numbers, discrete random variables, continuous random variables and random number generation. Techniques for generating random numbers, uniformity, independence, period, time, queueing models and random variate generation, inverse transform technique, empirical continuous distributions, empirical discrete distributions and discrete uniform distributions. Various simulation software tools (GPSS). Verification and validation of simulation models. Output Analysis.

BUSN461 STRATEGIC PLANNING AND MANAGEMENT

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. industry practices in the field of computer engineering for 30 working days.