1. Description of Courses (Sessions 2011 to 2014)

MA 100: Calculus 3(3,0)
Limits and continuity, derivatives, applications of derivatives. Integration, applications of integrals. Transcendental functions. integration techniques, L'Hopital's rule and improper integrals. Infinite Series.
Text: *Thomas's Calculus* by Finney, Weir and Giordano (10th ed.)

MA 101: Calculus 4(4,0)
Limits and continuity, derivatives, applications of derivatives. Integration, applications of integrals. Transcendental functions. integration techniques, L'Hopital's rule and improper integrals. Infinite Series.
Text: *Thomas's Calculus* by Finney, Weir and Giordano (10th ed.)

MA 103: Multivariate Calculus 3(3,0)
Multi-variable functions and their derivatives. Multiple integrals. Integration in Vector field. Vector differential calculus. Grad, Div, Curl.
Text: *Thomas's Calculus* by Finney, Weir and Giordano (10th ed.)

MA 202: Differential Equations and Transforms 4(4,0)
First order differential equations. Linear differential equations and second and higher order. Laplace transforms. Fourier series, integrals and transforms. Partial differential equations.
Text: *Advanced Engineering Mathematics* by Erwin Kreyszig (8th ed.).

MA 210: Linear Algebra and Complex Analysis 4(4,0)
Matrices, vectors, determinants, linear system of equations. Matrix eigenvalue problems. Complex numbers and functions. Complex integration. Power series, Taylor series. Laurent series, residue integration.
Text: *Advanced Engineering Mathematics* by Erwin Kreyszig (8th ed.).

MA 250: Probability and Statistics 3(3,0)
Data analysis, probability theory. Random variables. Probability distributions. Mathematical statistics, random sampling, estimation of parameters, confidence interval, hypothesis testing, goodness of fit, regression analysis, line fitting, correlation analysis.
Text: *Advanced Engineering Mathematics* by Erwin Kreyszig (8th ed.).

PHY 101: Mechanics and Wave Motion 4(3,1)
Measurement. Motion along a straight line. Vectors. Motion in 2 and 3 dimensions. Force and motion. Kinetic energy and work. Potential energy and conservation of energy. Center of mass and rotation. Center of mass and linear momentum. Torque and angular momentum. Equilibrium and elasticity. Gravitation. Fluids. Oscillations. Waves. First and second law of thermodynamics.
Text: *Fundamental of Physics* by Halliday, Resnick and Walker (Latest ed.)

PHY 103: Electricity and Magnetism 4(3,1)

Electric charge. Electric field. Gauss's law. Electric potential. Capacitance. Current and resistance. Circuits. Magnetic fields. Magnetic fields due to currents. Induction and inductance. Elctromagnetic oscillations and alternating current. Maxwell's equations.
Text: *Fundamental of Physics* by Halliday, Resnick and Walker (Latest ed.)

ME 201: Strength of Materials and Machine Design 3(3,0)
Topics include: physical and chemical properties of materials, motor, gears, levers, pressure and shafts.
Text: *A Text Book of Machine Design* by R. H. Hurmi

HU 101: Composition and Writing 3(3,0)
Principles of writing good English, understanding the composition process: writing clearly; word, sentence and paragraph. Comprehension and expression. Use of grammar and punctuation. Process of writing, observing, audience analysis, collecting, composing, drafting and revising, persuasive writing, reading skills, listening skills and comprehension, skills for taking notes in class, skills for exams.
Texts: *Ready to Write* by Karen Blanchard and Christine Root
Writing: Process, Product, and Power by Ken Davis and Kim Brian
Idea and Expression by Scott Foresman
Warriner's English Grammar and Composition, John E. Warriner

HU 202: Communication Skills 3(3,0)
Business communications; planning messages, writing concise but with impact. Letter formats, mechanics of business, letter writing, letters, memo and applications, summaries, proposals, writing resumes, styles and formats, oral communications, verbal and non-verbal communication, conducting meetings, small group communication, taking minutes. Presentation skills; presentation strategies, defining the objective, scope and audience of the presentation, material gathering material organization strategies, time management, opening and concluding, use of audio-visual aids, delivery and presentation.

Text: *Communication for Business* by Shirley Taylor

IS/HU 301: Language I 3(3,0)
Choice between Arabic, Chinese, French or German. Offering is subject to the availability of the Instructor. Purpose of this course is to give a working knowledge of a foreign language.
Text: *As prescribed.*

IS/HU 302: Language II 3(3,0)
Choice between Arabic, Chinese, French or German. Offering is subject to the availability of the Instructor. Follow up of Language I.
Text: *As prescribed.*

IS/HU 303: Language 3(3,0)
Choice between Arabic, Chinese, French or German. Offering is subject to the availability of the Instructor.
Text: *As prescribed.*

HU 312: Project Management 3(3,0)

To develop ability to plan and manage software development projects successfully, maximizing the return from each stage of the software development life cycle.

Texts: *Software Project Management,* Richard H. Thayer, Wiley IEEE Press 2002.
Software Engineering: A Practitioner's Approach, 4th edition, Roger S. Pressman, McGraw-Hill Higher Education.

HU 401: Entrepreneurship and Leadership 3(3,0)

The course focuses on the importance of entrepreneurship, in particular in the areas of leadership, decision making and negotiation Topics include: Leadership styles; group and organizational leadership; values and ethics; is leadership born or learned; Management vs. leadership; Entrepreneurship; Individual, group and organizational leadership principles; Human behavior and motivation in performance; Values and ethics in leadership and decision-making; Nature of entrepreneurial work - risks, rewards, challenges.

Text: *Entrepreneurship: A Contemporary Approach* by Donald F. Kuratko
The Art and Science of Leadership by Afsaneh Nahavandi, 2nd edition

EE 121: Circuit Analysis-I 4(3,1)

Electric quantities, electric signals, electric circuits, Kirchhoff's laws, circuit elements. Resistance, seres parallel combination, voltage and current dividers, resistive bridges and ladders, practical sources and loading, instrumentation and measurement. Nodal analysis, loop analysis, linearity and superposition, source transformation, One ports, circuit theorems,power calculations. Dependent sources, circuit analysis with dependent sources, ideal transformer, amplifiers. The operational amplifier, basic op-amp configurations, ideal op-amp circuit analysis, summing and difference amplifiers, amplifier types. Capacitance, inductance, natural response of RC and RL circuits. Response to DC forcing function. Transient response of first order circuits, step, pulse and pulse train responses, first order op-amp circuits. Transient response and step response of second order circuits.

Text: *Electric Circuit Fundamentals* by Sergio Franco (Latest ed.)

EE 222: Circuit Analysis - II 3(3,0)

Sinusoids and phasors, AC response of basic elements, first order circuits and second order circuits. Phasor algebra, AC impedance, Frequency domain analysis, op-amp AC circuits. AC power and 3-phase systems. AC Resonance. Laplace transform and its application to circuit analysis. Fourier Series analysis. Network functions, Bode plots.

Text: *Electric Circuit Fundamentals* by Sergio Franco (Latest ed.)

EE 223: Electronics-I 4(3,1)

Signals, frequency spectrum of signals, analog and digital signals. Amplifiers, voltage, current and power gains. Gains in decibels, amplifier power supplies, amplifier saturation, non-linear characteristics and biasing. Circuit models for amplifiers. Op-amp configurations and DC imperfections. Diodes, terminal characteristics, modeling of forward characteristics, operation in reverse region and breakdown. Rectifier circuits, limiting and clamping circuits, special diodes. MOSFETs, device structure and physical operation, current voltage characteristics, MOSFET circuits at DC. Biasing in MOSFET and small signal operation and model. BJTs, structure and physical operation, current voltage characteristics. BJT circuits at DC, biasing,

small signal operation and models, single stage amplifiers. Differential and multi stage amplifiers.
Text: *Microelectronic Circuits* by Sedra and Smith (latest ed.)

EE 321: Electronics-II 3(3,0)
Amplifiers, Operational Amplifiers, Feedback theory, Frequency Analysis, Advanced Operational Amplifiers with Applications, Voltage regulators, Oscillators, Special Electronic devices, Signal generation and wave shaping circuits. Output stages and power amplifiers.
Texts: *Microelectronic Circuits* by Sedra and Smith (latest ed.)
Electronic Devices and Circuits, Theodore Bogart (6th ed.)

EE 322: Electronics Design Laboratory 1(0,1)
This laboratory is design oriented where the students are required to design and deliver small projects related to the concepts and theory studied during Electronics I and Electronics II courses.
Reference Material: *Electronic Devices and Circuits* by Theodore Bogart (6th ed.), E*lectronics II Lab Manual*

EE 323: Electronics-II 4(3,1)
Amplifiers, Operational Amplifiers, Feedback theory, Frequency Analysis, Advanced Operational Amplifiers with Applications, Voltage regulators, Oscillators, Special Electronic devices, Signal generation and wave shaping circuits. Output stages and power amplifiers.
The course is coupled with a design oriented laboratory wherein the students carry out tasks assigned to them involving design of small projects.
Texts: *Microelectronic Circuits* by Sedra and Smith (latest ed.)
Electronic Devices and Circuits, Theodore Bogart (6th ed.)
Reference Material: *Electronics II Lab Manual*

CS 100: Introduction to Computing 2(1,1)
This broad-based, entry-level course provides a general overview of the main concepts in computing, Topics studied include the history and evolution of computers, central processing unit, data storage, input/output devices, multimedia, operating systems, programming languages, networking, the Internet, systems analysis and design, management information systems, electronic commerce, security and privacy issues, ethical issues and the computing profession. Students will become familiar with popular operating systems, software applications and tools, and learn the basic concepts of programming by developing interactive web pages.
Text: *Computers: Tools for an Information Age*, 8th ed., H.L. Capron, Addison Wesley, 2003.

CS 101: Introduction to Computing 3(2,1)
This broad-based, entry-level course provides a general overview of the main concepts in computing, Topics studied include the history and evolution of computers, central processing unit, data storage, input/output devices, multimedia, operating systems, programming languages, networking, the Internet, systems analysis and design, management information systems, electronic commerce, security and privacy issues, ethical issues and the computing profession. Students will become familiar with popular operating systems, software

applications and tools, and learn the basic concepts of programming by developing interactive web pages.
Text: *Computers: Tools for an Information Age*, 8th ed., H.L. Capron, Addison Wesley, 2003.

CSE 120: Digital Logic Design 4(3,1)
Design concepts. Introduction to logic circuits, truth tables, logic gates and networks, boolean algebra, synthesis using AND, OR and NOT gates, NAND and NOR logic networks, design examples, introduction to CAD tools, introduction to Verilog. Implementation technology. Optimized implementation of logic functions. Number representation and arithmetic circuits. Combinational circuit building blocks. Flip flops, registers, counters and simple processor.
Text: *Fundamentals of Digital Logic with Verilog Design* by Brown and Vranesic (Latest ed.)

CS 140: Programming Fundamentals I 4(2,2)
This course is designed to train the fresh incoming bachelor students of computer science and computer engineering students to solve problems through digital computers. Students learn to solve mathematical and scientific problems by developing computer programs. They learn conversion of problem solving techniques to algorithms and flowcharts. They view solution as 'data + algorithm = program'. They learn representing information in variables of primitive data types of C language and practice control constructs (if-else, switch, for, while, do-while) to convert the information space i.e., variables such that solution is available in the space. Collection data constructs like structures, arrays and strings are then taught to enrich problem state representation. After learning structured programming the students practice functions in modular programming mode.
Text: *Theory and Problems of Programming with C* by Bryon S. Gottfried, 2nd Ed. McGraw Hill

CS 141: Programming Fundamentals 3(2,1)
The course is designed to provide the students an overview of the fundamentals of programming techniques and a high level insight into computer systems. It focuses on developing problem solving approach in the students and concentrates on formulation of algorithms and translating them into structured programs.
Text: *Deitel & Deitel, C++ How to Program,* 4th Edition, Prentice Hall

CS 142: Programming Fundamentals II 3(2,1)
The course is an introduction to computer science. This course could be taught in languages like Python or Scheme. The course naturally divides itself into four parts: the first two parts are on building procedural and data abstractions, the third part is on building new programming languages, and the last part is an optional "topics" part. The first three parts mainly focus on techniques for managing large programmes: this includes various methods of abstraction and designing new computer languages. The first part starts with functions, control of flow, recursion and higher order functions. This part is an introduction to building abstractions with functions. The second part of the course then goes on to build abstractions with data. This leads us naturally to object-oriented programming. A third method of controlling complexity is to define new languages and build interpreters. This is the focus of the third part of the course. There are two clear independent directions in the last part of the course: either the course can introduce the ideas of processing large amounts of data or it can go ahead and implement a language on a register machine.

Texts: *Structure and interpretation of computer programs* by Abelson, Sussman and Sussman. http://mitpress.mit.edu/sicp/
Composing programs by John DeNero. www.composingprograms.com

References: *Python programming*, for the absolute beginner, by Michael Dawson, 3rd edition. Course Technology.
Dive into Python 3 by Mark Pilgrim. 2nd edition.
Learning Python by Mark Lutz, 5th edition. O'Reilly
The Scheme programming language by Kent Dybvig. 4th edition .
Simply Scheme by Brian Harvey. 2nd edition.

CS 200: Discrete Mathematics 3(3,0)

This course offers an intensive introduction to discrete mathematics as it is used in computer science. Topics include functions, relations, sets, series and sequences, simple proof techniques, boolean algebra, propositional logic, digital logic, elementary number theory, fundamentals of counting, predicate logic, recurrence relations, graphs, trees, matrices, computational complexity, elementary computability, and discrete probability.

Texts: *Discrete Mathematics and its Applications*, Kenneth H. Rosen, 5th Edition, 2003, McGraw-Hill.
Discrete Mathemactis with Applications, Sussana S. Epp. 4th Edition, 2011, Brooks/Cole.

CS 202: Theory of Automata and Formal Languages 3(3,0)

This course covers foundational theory and practice of finite state machines, regular expression matching, and context-free grammars. The following are the course learning objectives:

a) design of finite state automata and regular expressions
b) prove that a language is not regular
c) design of push-down automata and context-free grammars
d) conversion amongst push-down automata and context-free grammars
e) prove that a language is not context-free

Through taking this course, students will learn the theoretical and practical significance of automata theory and its application to important real-world problem domains, such as parsing, programming language design, security policy specification, natural language processing, and many others

Texts: *Introduction to Computer Theory,* Daniel I. A. Cohen 2nd Edition.
Introduction to Automata Theory, Languages, and Computation, John E. Hopcroft. Jeffery D. Ullman, Addison-Wesley. ISBN 0-201-02988

CS 210: Data Structures and Algorithms 4(3,1)

This course covers basic data arrangement and manipulation techniques. Students learn different datum/object collection constructs to different problem representation in OO models of stack, queue, priority queue, list, doubly linked list, binary trees, m-ary trees, and graphs along with their mutation functions. These structures are practiced through dynamic memory manipulations. Searching, sorting and hashing techniques are also taught and practiced in the laboratory exercises. File processing is extensively involved to solve complex problems.

Text: *Data Structures and Algorithm Analysis*, M. A. Weiss, Pearson, 2nd Edition, 1997.

CSE 225: Computer Organization and Assembly Language 4(3,1)

This course will provide an overview of the structure and function of computers at the register level and how data is represented at this level. Main topics of the course are: organization of computer hardware, the functions of assembler, linker, and loader, basic assembly language instruction set, memory management, addressing modes, RISC and CISC comparison, macros, file handling, real and protected mode interrupts, stack and procedures, low-level I/O. Students will learn to design, implement, and debug programs in assembly language.

Texts: *Assembly Language Programming and Organization of IBM PC* by Ytha Yu and Charles Marut, McGraw- Hill, 1992

Computer Organization and Design, David A. Patterson, John L. Hennessy, 3rd Edition. *Assembly Language Programming and Organization of IBM PC*, Ytha Yu and Charles

Introduction to Assembly Language Programming for Pentium and RISC Processors, Sivarama P. Dandamudi

CS 241: Object-Oriented Programming 3(2,0)

This course is designed to teach object-oriented programming techniques to those who have learned basic programming concepts. Concepts include: Classes; Objects; Data Protection, Data Abstraction; Data Encapsulation; Language extension through operator overloading; Template functions and Function Overloading; Friend Functions; Association, aggregation, composition of objects; Inheritance of object structure and object knowledge. Overridden Methods; Virtual Functions; Pure Virtual Functions: Concrete and Abstract Classes; Polymorphism in objects related through is-a relationships of inheritance; Class Templates; and Files. C++ programming language is used to demonstrate the object-oriented programming constructs.

Texts: *C++ How to Program*, Deitel & Deitel,4th Edition, Prentice Hall

C++ Programmer's Notebook, Jim Keogh, 2nd Edition.

CSE 301: Control Systems 4(3,1)

The course familiarizes the student with the mathematical tools and skills needed to analyze and design feedback control systems. Upon completion of this course, students will be able to create mathematical models using block and signal flow diagrams, become familiar with Root Locus techniques, and be able to sketch Root Locus and Bode plots. Design basic feedback systems.

Texts: *Modern Control Engineering* by Katsuhiko Ogata, Pearson Education, 4th Edition, 2002.

Control Systems Engineering by Norman S. Nise, John Wiley, Asia, 6th Edition.

CS 310: Analysis of Algorithms 3(3,0)

This course covers a number of ideas and techniques useful for designing and analyzing data structures and algorithms. In particular, we will introduce techniques for analyzing upper bounds for algorithms and lower bounds for problems. Problem areas include sorting, graphs, dynamic programming, combinatorial algorithms, computational geometry, encryption, and NP-Completeness.

Text: *Introduction to Algorithms,* T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. D. Stein, MIT Press.

CSE 311: Numerical Analysis and Computing 3(3,0)

This course makes the students understand the basic techniques of numerical analysis for solving nonlinear algebraic equations, interpolation, for numerical differentiation and

integration. It enables them to set up computational algorithms for the solution of above problems.
Text: *A First Course in Numerical Analysis,* Saeed Bhatti & Naeem Bhatti, 4th Edition, 2003.

CSE 320: Embedded Systems Development 4(3,1)
Introduction, design challenges, process technologies, custom single purpose processor design, general purpose processors (software), development environment, debuggers, emulators, standard off the shelf processors, peripherals, memory, interfacing, digital camera design example, state machine and concurrent process model, design technology, PLA, FPGA introduction, hardware/firmware partitioning, circuit debugging, firmware design, firmware architecture.
Texts: *Embedded Systems Design: An Introduction to Processes, Tools, and Techniques* by Arnold S. Berger.
Embedded System Design : A Unified Hardware and Software Approach, Tony Vahid and Givargis

CS 325: Operating Systems 3(3,0)
This is designed to be the first undergraduate level course with the main objective of teaching students the concepts and principles that underlie the design and implementation of contemporary operating systems. The discussion of the concepts and principles used in UNIX, Linux, and MS-DOS/MS-Windows operating systems will be integrated throughout the course. Students will be given the opportunity to examine important UNIX and Linux data structures and use its various services through programming exercises. Programming language environment: C on UNIX/Linux.
Texts: *Operating System Concepts,* Silberschatz, Galvin and Gagne, 6th Edition, 2002, John Wiley & Sons, Inc.
Modern Operating Systems, Tanenbaum A.S., 2nd Edition, 2001

CS 326: Operating Systems 3(2,1)
This is designed to be the first undergraduate level course with the main objective of teaching students the concepts and principles that underlie the design and implementation of contemporary operating systems. The discussion of the concepts and principles used in UNIX, Linux, and MS-DOS/MS-Windows operating systems will be integrated throughout the course. Students will be given the opportunity to examine important UNIX and Linux data structures and use its various services through programming exercises. Programming language environment: C on UNIX/Linux.
Text: *Modern Operating Systems*, Tanenbaum A.S., 2nd Edition, 2001

CS 342: Web Technologies 3(3,0)
This course will cover many of new and upcoming technology; Student will get hands on practices on different technologies; this course covers Client Side Technologies; HMTL, JavaScript, AJAX and Server Side Technologies; PHP, PHP, Web Services and Rich Internet Application Framework; FLEX Also they will get introduction to upcoming technologies like Semantic Web
Text: This course is a dynamic course being updated each year. Teachers are encouraged to prescribe latest books and material.

CS 360: Artificial Intelligence 4(3,1)

This is an introductory course in Artificial Intelligence. This course introduces representations, techniques, and architectures used to build applied systems and to account for intelligence from a computational point of view. This course also explores applications of rule chaining, heuristic search, logic, constraint propagation, constrained search, and other problem-solving paradigms. In addition, it covers applications of decision trees, neural nets, SVMs and other learning paradigms. The essentials of Prolog and an overview to its applications.

Texts: *Expert Systems and Applied Artificial Intelligence* by Efraim Turban
Artificial intelligence: a modern approach. Russell, Stuart J., and Peter Norvig. 2nd edition. Upper Saddle River, NJ: Prentice Hall, 2003. ISBN: 0137903952.
Artificial Intelligence by George F. Luger

CS 370: Database Systems 4(3,1)

The course discusses the fundamental concepts of database design and implementation focusing primarily on relational data model. Structured Query Language (SQL) is also introduced and practiced in the course.

Text: *Fundamentals of Database Systems,* Elmasri & Navathe, 4th Edition.

CSE 371: Signals and Systems 3(3,0)

The course presents and integrates the basic concepts for both continuous-time and discrete-time signals and systems. Signal and system representations are developed for both time and frequency domains. These representations are related through the Fourier transform and its generalizations, which are explored in detail.

Texts: *Linear Systems and Signals* by B. P. Lathi (Latest ed.)
Signals and Systems by Oppenheim, Willsky and Nawab, 2nd ed., Pearson Education Ltd., (Discrete Systems and DTFT)
Analog and Digital Communication Systems by B. P. Lathi and Z. Ding, Oxford Univ. Press, 2009. (for Fourier Transform)

CS 375: Data Mining 3(3,0)

Data Mining and Knowledge Discovery studies how to analyze the flood of information generated by businesses, science, web, and other sources. It uses methods from several fields, including databases, machine learning, statistics, and information visualization and it focuses on key tasks such as classification, clustering, market basket analysis or association rules, and link analysis.

Texts: *Data Mining: Practical machine learning tools and techniques* by Witten, I. H., & Frank, Morgan Kaufmann.
Data Mining: concepts and techniques by Han, J., & Kamber, M Morgan Kaufmann

CS 390: Software Engineering 4(3,1)

This course is designed to provide understanding of the concepts, techniques and tools for the definition, design, development, testing and maintenance of software systems. The main objective is to emphasize the need for a systematic and managed approach to software development. The course discusses important software development activities, emphasizing on analysis, design, and testing.

Text: *Software Engineering*, Ian Sommerville, 6th Edition, Addison-Wesley, 2001.

CS 391: Object Oriented Analysis and Design 3(3,0)
This course requires prior experience in object-oriented programming and familiarity with basic object-oriented concepts such as abstraction, encapsulation, inheritance, composition and aggregation. Object-oriented analysis and design concepts are introduced using UML, and an iterative evolutionary design process based on the Unified Process applied in an agile spirit. Students will also learn applications of object-oriented design principles and patterns such as GRASP and GoF design patterns.
Text: *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development*, Craig Larman, 3rd edition, Prentice Hall, 2005 (or Latest edition).

CSE 400: Industrial Automation 4(3,1)
The course offers instruction in electronics applications and computer aspects of automation, including both theory and applications of digital and industrial electronics, hydraulics/pneumatics, actuators, transducers and robotic systems. It also includes discussion of sensors, transducers, signal conditioning devices and computer based data acquisition. Programmable Logic Controllers (PLC) and Ladder Logic are also covered.
Text: *Industrial Control Electronics: Devices, Systems & Applications,* Terry L. M. Bartlet, 3rd Edition, June 2005.

CSE 420: Computer Architecture 3(3,0)
This course focuses on the principles, practices and issues in computer architecture, while examining computer design trade-offs both qualitatively and quantitatively. Topics include: Advanced Pipelining, Instruction-level parallelism, Memory - hierarchy design, Storage Systems and input/ output and Multiprocessor architecture
Text: *Computer Architecture: A Quantitative Approach,* Hennessy, J. L, and Patterson, D. A., Latest edition.

CSE 421: Digital Design 4(3,1)
The course teaches the design and implementation of digital systems using EDA tools and FPGA. Upon completion of this course, students will have a good comprehension of EDA tool XILINX, become familiar with XILINX FPGA and its programming, programming of Programmable Logic Array and CPLD.
Text: *Advanced Digital Design with the Verilog HDL* by Michael D. Ciletti, Prentice Hall, 2003.

CSE 422: Computer Architecture 3(2,1)
This course focuses on the principles, practices and issues in computer architecture, while examining computer design trade-offs both qualitatively and quantitatively. Topics include: Advanced Pipelining, Instruction-level parallelism, Memory - hierarchy design, Storage Systems and input/ output and Multiprocessor architecture
Text: *Computer Architecture: A Quantitative Approach,* Hennessy, J. L, and Patterson, D. A., Latest edition.

CSE 430: Wireless Networks 3(3,0)
The course deals with the fundamental and practical aspects in the analysis and design of wireless systems. Topics that will be covered are: the wireless communication channel, spectrum efficient digital modulation techniques their comparison and detection techniques,

cellular communication principles, techniques used to combat the channel, overview of multiple access techniques and example wireless communication systems.
Text: *Introduction to Wireless Systems* by P. M. Shankar. John Wiley & Sons, 2002.

CSE 471: Communication Systems and Wireless 3(3,0)
The course covers basic principles of communication theory with emphasis on analog modulation systems. Upon completion of this course, students will become familiar with the fundamental features and basic building blocks of analog and digital communications systems, and also be able to characterize the performance of analog modulation techniques in noise
Texts: *Modern Digital and Analog Communication System*s by B. P. Lathi, 3rd edition, Oxford University Press.
Introduction to Wireless Systems by P. M. Shankar. John Wiley & Sons, 2002.

CSE 472: Digital Signal Processing 3(3,0)
The course begins with a discussion of the analysis and representation of discrete-time signals and systems including a discussion of discrete-time convolution, difference equations, the z-transform and the discrete Fourier transform. Considerable emphasis is placed on the similarities with and distinctions between discrete-time and continuous-time signals and systems. The course then proceeds to a consideration of digital network structures for implementation of both recursive (infinite impulse response) and non-recursive (finite impulse response) digital filters. A major consideration in digital signal processing is the design of digital filters to meet prescribed specifications. Thus a set of lectures is devoted to a detailed discussion of digital filter design for both recursive and non-recursive filters. The course concludes with a thorough presentation of the fast Fourier transform algorithm for computation of the discrete Fourier transform.
Texts: *Discrete Time Signal Processing* by Oppenheim, Alan V., and Ronald W. Schafer, 3rd Edition, Prentice Hall, 2010.
A Course in Digital Signal Processing, Boaz Porat, John Wiley & Sons, Latest ed.

CSE 491: Final Project 1 3(0,3)
The final project involves research, planning and development of a real and substantial project related to computer science and engineering. It provides an opportunity to the students to use their acquired academic skills in the form of a demonstrable software product. Make oral and written project presentations.

CSE 492: Final Project 1I 3(0,3)
The final project involves research, planning and development of a real and substantial project related to computer science and engineering. It provides an opportunity to the students to use their acquired academic skills in the form of a demonstrable software product, make oral and written project presentations.