5 year Integrated M.Sc. (Computer Science)

(Exit option after 3 years with B.Sc. (Hons) degree in Computer Science)

CURRICULUM

(w.e.f. 2017-18)

DEPARTMENT OF COMPUTER SCIENCE

SCHOOL OF PHYSICAL SCIENCES

DOON UNIVERSITY

Course Structure

SEMESTER I						
Course Type	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
Core	CSC-101	Introduction to Computer Science Using Python	4		2	6
Core	CSC-102	Digital System Design	4		2	6
Core	CSC-103	Basic Mathematics	3	1		4
AECC	EES-110	Environmental Studies	2			2
SEC	CSS-100	Introduction to the IoT	1		1	2
Generic	CSG-101	Introduction to Computer Science Using Python	4		2	6
	CSG-102	Digital System Design	5	1		6
	CSG-103	Basic Electronics	5	1		6
					Total Credits	**26**
SEMESTER II						
Course Type	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
Core	CSC-151	Programming in C	4		2	6
Core	CSC-152	Computer Architecture	4		2	6
Core	CSC-153	Discrete Mathematics	3	1		4
Core	CSC-154	Project-1: Embedded System Development				2
AECC	ENG-151	English	2			2
SEC	CSS-150	Unix System Administration & Shell Programming			2	2
Generic	CSG-151	Programming in C	4		2	6
	CSG-152	Computer Architecture	5	1		6
	CSG-153	Embedded System Design	5	1		6
					Total Credits	**28**
SEMESTER III						
Course Type	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
Core	CSC-201	Object Oriented Programming Using C++	4		2	6
Core	CSC-202	Operating Systems	4		2	6
Core	CSC-203	Data Structures	4		2	6
SEC	CSS-200	Client-side Web Technologies			2	2
Generic	CSG-201	Object Oriented Programming Using C++	4		2	6
	CSG-202	Operating Systems	5	1		6
	CSG-203	Econometrics	5	1		6
					Total Credits	**26**

SEMESTER IV						
Course Type	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
Core	CSC-251	Programming in Java	4		2	6
Core	CSC-252	Database Systems	4		2	6
Core	CSC-253	Design and Analysis of Algorithms	4		2	6
Core	CSC-254	Project-2: Web Application Development				2
SEC	CSS-250	Server-side Web Technologies			2	2
Generic	CSG-251	Programming in Java	4		2	6
	CSG-252	Database Systems	4		2	6
	CSG-253	Mathematical Methods	5	1		6
					Total Credits	**28**
SEMESTER V						
Course Type	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
Core	CSC-301	Automata Theory and Computability	3	1		4
Core	CSC-302	Computer Networks	4		1	5
Core	CSC-303	Numerical and Statistical Computing	4		1	5
Core	CSC-304	Software Engineering	3	1		4
DSE-1	CSE-301	Modeling and Simulation	3		1	4
	CSE-302	Operations Research	3	1		4
	CSE-303	Biometrics	3	1		4
					Total Credits	**22**
SEMESTER VI						
Course Type	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
Core	CSC-351	Artificial Intelligence	4		1	5
Core	CSC-352	Computer Graphics	3		1	4
Core	CSC-353	Optimization Techniques	4		1	5
Core	CSE-354	Project Work/Dissertation				6
DSE-2	CSE-351	Natural Language Processing	3		1	4
	CSE-352	Digital Marketing	3		1	4
	CSE-353	Introduction to Data Science	3		1	4
					Total Credits	**24**

SEMESTER VII						
Course Type	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
Core	CSC-401	Probability and Statistical Inference	4		2	6
Core	CSC-402	Compiler Design	4		2	6
DSE-3,4	CSE-401	Computational Number Theory	3		1	4
	CSE-402	Digital Signal Processing	3		1	4
	CSE-403	Data Mining and Warehousing	3		1	4
	CSE-404	Fuzzy Logic	3		1	4
	CSE-405	Wireless and Mobile Computing	3		1	4
	CSE-406	Introduction to Cloud Computing	3		1	4
Self-Reading	CSR-400	Logic for Computer Science				2
Presentation	CSP-400	Research Seminar-1				2
					Total Credits	**24**
SEMESTER VIII						
Course Type	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
Core	CSC-451	Advanced Algorithms	4		2	6
Core	CSC-452	Information Theory and Coding	5	1		6
DSE-5,6	CSE-451	Cryptography and Network Security	3		1	4
	CSE-452	Digital Image Processing	3		1	4
	CSE-453	Machine Learning	3		1	4
	CSE-454	Neural Networks	3		1	4
	CSE-455	Mobile Ad-hoc Networks	3		1	4
	CSE-456	Cloud Architecture	3	1		4
Self-Reading	CSR-450	Knowledge Representation				2
Presentation	CSP-450	Research Seminar-2				2
					Total Credits	**24**

SEMESTER IX						
Course Type	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
Core	CSC-501	Parallel and Distributed Computing	4		2	6
Core	CSC-502	Combinatorics and Graph Theory	5	1		6
Core	CSC-503	Project				4
DSE-7,8	CSE-501	Cyber Forensics	3		1	4
	CSE-502	Computer Vision and Pattern Recognition	3		1	4
	CSE-503	Big Data Analytics	3		1	4
	CSE-504	Evolutionary Algorithms	3		1	4
	CSE-505	Software Defined Networks	3		1	4
	CSE-506	Cloud Security	3	1		4
					Total Credits	**24**
SEMESTER X						
Course Type	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
Core	CSC-551	Research Methodology	4			4
Core	CSC-552	Master’s Thesis				18
					Total Credits	**22**

Guidelines

1. Minimum Credit Requirements

For 3 year B.Sc. (Hons) in Computer Science

Course Type	**Number of papers**	**Credits**
Core	22 papers	110
AECC (Ability Enhancement Compulsory Course)	2 papers of 2 credits each	2 × 2 = 4
SEC (Skill Enhancement Course)	4 papers of 2 credits each	4 × 2 = 8
Generic	4 papers of 6 credits each	4 × 6 = 24
DSE (Discipline Specific Elective)	2 papers of 4 credits each	2 × 4 = 8
	Total Credits	**154**

For 5 year Integrated M.Sc. (Computer Science)

Course Type	**Number of papers**	**Credits**
Core	31 papers	172
AECC (Ability Enhancement Compulsory Course)	2 papers of 2 credits each	2 × 2 = 4
SEC (Skill Enhancement Course)	4 papers of 2 credits each	4 × 2 = 8
Generic	4 papers of 6 credits each	4 × 6 = 24
DSE (Discipline Specific Elective)	8 papers of 4 credits each	8 × 4 = 32
Self-Reading	2 papers of 2 credits each	2 × 2 = 4
Presentation (Research Seminar)	2 papers of 2 credits each	2 × 2 = 4
	Total Credits	**248**

A minimum of 154 credits and 248 credits respectively must be earned by the student for the award of B.Sc. (Hons) and M.Sc. in Computer Science degrees.

2. Generic Courses

Student must take 4 generic papers, one each in semesters I through IV. They can opt for papers from following disciplines.

- Physics
- Electronics
- Mathematics
- Statistics
- Operations Research
- Economics
- Commerce

Following is the list of generic electives offered by the department:

SEMESTER I	SEMESTER II	SEMESTER III	SEMESTER IV
CSG-101: Introduction to Computer Science Using Python	CSG-151: Programming in C	CSG-201: Object Oriented Programming Using C++	CSG-251: Programming in Java
CSG-102: Digital System Design	CSG-152: Computer Architecture	CSG-202: Operating Systems	CSG-252: Database Systems
CSG-103: Basic Electronics	CSG-153: Embedded System Design	CSG-203: Econometrics	CSG-253: Mathematical Methods

3. Projects

Traditional teaching, usually based on lectures and tutorials, fosters the idea of instruction-driven learning model where students are passive listeners. Besides this approach, project based learning as a different learning paradigm is standing behind constructivism learning theory, where learning from real-world situations is put on the first place.

There are 2 mini projects in II and IV semesters and a major project in the VI semester. A brief outline about the projects is mentioned below:

- Project-1: Embedded System Development

  The focus of this project lies in designing a system/device for a real-world application. The skills required to complete this project are basic knowledge of IoT and Embedded System with some C/Python programming. All of the requirements are covered in the first year of the curriculum.

- Project-2: Web Application Development

  In this project, students are expected to work in team and apply all their learning from the courses like web technologies, databases etc. (covered in the second year of the curriculum) into developing a Web Application.

- Project Work/Dissertation

  This course requires individual/group effort that is overseen by your project supervisor. Weekly meetings will be held to discuss progress and review necessary documents in support of the project. There will be monthly presentations regarding progress of the project. A final presentation followed by viva-voce by external examiner will be held at the end of the semester where the student must submit a project report.

4. Self-Reading Courses

One Self-Reading course is included in each of VII and VIII. The goal of this initiative is to inculcate self-learning among the students. The offering will involve student presentations, group discussions and tests. To complete such courses, students will be provided necessary resources such as study material, online links etc. by the course instructor.

5. Research Seminar

The Research Seminars will be conducted in VII and VIII semesters with the main objective of building up research skills in the students. Under this course module, the students must present quality research papers relating to the electives (DSE) chosen by them in the corresponding semester and explore the significance of the selected paper.

Learning Outcomes

1. Program Specific Outcomes

B.Sc. (Honours) Computer Science

PO1. Demonstrate the aptitude of computer programming and computational problem solving skills.
PO2. Develop proficiency in the practice of computing.
PO3. Ability to apply computer science theory, mathematical foundations, and algorithmic principles in modeling, designing and implementation of computer-based systems (process, component, or program) to solve real world problems.
PO4. Ability to appreciate and use current emerging technologies, skills and tools necessary for computing practice.
PO5. Ability to pursue higher studies of specialization and to take up technical employment.

M.Sc. Computer Science

PO1. Communicate computer science concepts, designs, and solutions effectively and professionally.
PO2. Apply computer science theory and software development concepts to produce effective designs and solutions for specific problems.
PO3. Identify, analyze, and synthesize scholarly literature relating to the field of computer science.

2. Course Specific Outcomes

Course Code and Title	**Course Outcome** (After completing the course, the students will be able to)
SEMESTER I	
CSC-101: Introduction to Computer Science Using Python	CO1. Understand basic principles of Python programming language. CO2. Implement object oriented concepts. CO3. Implement database and GUI applications.
CSC-102: Digital System Design	CO1. Understand different number systems and their conversions. CO2. Understand difference between digital vs. analog signal and various digital components of computer hardware, which we use on daily purpose. CO3. Analyze and minimize Boolean expressions. CO4. Analyze, design and implement combinational and sequential circuits. CO5. Understand basic concepts of VHDL. CO6. Explain mathematical and physical model of every digital electronics component.

CSC-103: Basic Mathematics	CO1. Demonstrate a working knowledge of mathematical application of given values and given vectors in a variety of applied fields. CO2. Demonstrate a working knowledge of Taylors Theorem and its application. CO3. Model physical systems using Differential equations.
EES-110: Environmental Studies	CO1. Creating the awareness and imparting basic knowledge about the environment and its allied problems. CO2. Developing an attitude and motivating students to participate in environment protection and environment improvement. CO3. Striving to attain harmony with environment.
CSS-100: Introduction to the IoT	CO1. Understand latest technology "Internet of Things", which is used worldwide nowadays. CO2. Evolution and architecture of IoT is explained in terms of hardware, programming language and its wide spectrum applications. e.g. UBER, OLA cabs
SEMESTER II	
CSC-151: Programming in C	CO1. Understand program flow using conditional and iterative statements. CO2. Understand arrays, pointers, structures and unions. CO3. Define functions that can receive variables, arrays, pointers and structures. CO4. Create open, read, manipulate, write and close files.
CSC-152: Computer Architecture	CO1. Understand the structure, function and characteristics of computer systems. CO2. Understand the design of the various functional units and components of computers. CO3. Identify the elements of modern instructions sets and their impact on processor design. CO4. Explain the function of each element of a memory hierarchy and computer I/O.
CSC-153: Discrete Mathematics	CO1. Learn propositional and predicate logic, thereby developing mathematical thinking and logical reasoning ability. CO2. Express statements written in natural language using logics. CO3. Understand set theory, relations, functions, and mathematical proofs. CO4. Apply counting principles to solve various combinational problems. CO5. Formulate and solve recurrence relations for recursive solutions. CO6. Explain elementary group theory. CO7. Understand graph, their types and model problems using graphs.

CSC-154: Project-1: Embedded System Development	CO1. Formulate a problem statement based on the task requirements. CO2. Design and develop an embedded system to solve the task using microcontrollers like Arduino, Raspberry Pi and various sensors. CO3. Write a technical report and demonstrate the project.
ENG-151: English	CO1. Correct usage of English grammar in communication (writing and speaking). CO2. Learn to write letters, essays and summaries. CO3. Speak in English with improved fluency and comprehensibility.
CSS-150: Unix System Administration & Shell Programming	CO1. Understand the architecture, networking and basic commands of UNIX. CO2. Identify and use UNIX/Linux utilities to create and manage simple file processing operations, organize directory structures with appropriate security. CO3. Implement various file processing commands used in UNIX. CO4. Develop shell scripts for simple applications.
SEMESTER III	
CSC-201: Object Oriented Programming Using C++	CO1. Learn various OOPs concepts and use them to solve problems. CO2. Understand inheritance, polymorphism, type casting, excepting handling. CO3. Apply good programming principles to write C++ programs.
CSC-202: Operating Systems	CO1. Explain the objectives and functions of modern operating systems. CO2. Understand the process management, scheduling and IPC. CO3. Understand the working of an OS as a resource manager, file system manager, process manager, memory manager and I/O manager and methods used to implement the different parts of OS.
CSC-203: Data Structures	CO1. Differentiate between data structures and ADTs. CO2. Efficiently implement various linear and non-linear data structures – stack, queue, priority queue, trees, graphs. CO3. Select and use appropriate data structures to solve a problem.
CSS-200: Client-side Web Technologies	CO1. Create websites using HTML and CSS, targeted for viewing on multiple platforms (mobile and desktop). CO2. Implement client-side programming using JavaScript. CO3. Programmatically manipulate CSS and the Document Object Model to implement client-side behaviors and visual effects.

SEMESTER IV	
CSC-251: Programming in Java	CO1. Learn basics of Java class, objects, methods, constructors, arrays, strings. CO2. Learn interfaces, exception handing in Java. CO3. Learn applets, and advanced topics.
CSC-252: Database Systems	CO1. Differentiate between File system and DBMS. CO2. Develop ER diagram for representing conceptual data model. CO3. Convert ER diagram into a set of relations representing logical data model and formulate relational algebra queries. CO4. Write Structured Query Language (SQL) to interact with DB. CO5. Understand transaction and concurrency control.
CSC-253: Design and Analysis of Algorithms	CO1. Understand concept of asymptotic analysis and perform complexity analysis of iterative and recursive algorithms. CO2. Formulate and solve time complexity recurrence relations using various techniques. CO3. Solve computational problems using various algorithmic paradigms like divide-and-conquer, greedy, dynamic programming, backtracking, branch-and-bound. CO4. Understand different classes of problems like P, NP, NP-Hard and NP-Complete.
CSC-254: Project 2 Web Application Development	CO1. Develop skills in client-side web application development technologies including HTML, CSS and Javascript. CO2. Determine the needs for web database and connectivity. CO3. Develop a functional web application.
CSS-250: Server-side Web Technologies	CO1. Develop web based application using suitable client side and server side web technologies. CO2. Connect a java program to a DBMS and perform insert, update and delete operations on DBMS table and also write a server side java application called Servlet to catch form data sent from client, process it and store it on database. CO3. Have a good grounding of web application terminologies, Internet tools, e-commerce and other web services.
CSG-253: Mathematical Methods	CO1. Demonstrate a working knowledge of mathematical application of Fourier transforms, Inverse Fourier transforms. CO2. Use of PDE and applications in solving heat equation and wave equation. CO3. To find power series solutions of differential equation.

SEMESTER V	
CSC-301: Automata Theory and Computability	CO1. Learn about formal languages and Chomsky hierarchy. CO2. Understand pumping lemma for regular and context free languages. CO3. Learn about regular grammar, DFA, NFA, Mealy and Moore machine, PDA, Turing machines. CO4. Understand the notion of decidability and computability.
CSC-302: Computer Networks	CO1. Recognize the technological trends of Computer Networking. CO2. Discuss the key technological components of the Network. CO3. Evaluate the challenges in building networks and solutions to those.
CSC-303: Numerical and Statistical Computing	CO1. Able to solve non linear equations using differential numerical method. CO2. Understand floating point representation and arithmetic. CO3. Use of C programming to solve interpolation problems.
CSC-304: Software Engineering	CO1. Apply the software engineering lifecycle by demonstrating competence in communication, planning, analysis, design, construction, and deployment. CO2. Work as an individual and as part of multidisciplinary team to develop and deliver quality software CO3. Demonstrate and apply current theories, models, and techniques that provide a basis for the software lifecycle.
CSE-302: Operations Research	CO1. Able to select appropriate goal and value of different variables. CO2. Formulate an appropriate model of the problem. CO3. Design new simple models like CPM to improve decision making and develop critical thinking and objective analysis of decision problem.
SEMESTER VI	
CSC-351: Artificial Intelligence	CO1. Analyze and formalize the problem as a state space, graph, and design heuristics. CO2. Apply knowledge representation techniques and problem solving strategies to common AI applications. CO3. Learn about expert systems, planning and learning in AI. CO4. Understand the numerous applications and huge possibilities in the field of AI.
CSC-352: Computer Graphics	CO1. Implement various geometric algorithms, transformations, area filling, clipping. CO2. Describe the importance of viewing and projections. CO3. Define the fundamentals of animation, virtual reality and related technologies. CO4. Apply mathematics and logic to develop programs for elementary graphic operations.

CSC-353: Optimization Techniques	CO1. Understand the theory of optimization methods and algorithms developed for solving various types of optimization problems. CO2. Model engineering minima/ maxima problem as optimization problems. CO3. Use Matlab to implement optimization algorithms.
CSC-354: Project/ Dissertation	CO1. Identify and define the problem statement. CO2. Define and justify scope of the proposed problem. CO3. Gather and analyze system requirements. CO4. Propose an efficient solution. CO5. Develop technical report writing and oral presentation skills.
CSE-352: Digital Marketing	CO1. Learn Search Engine Optimization (SEO). CO2. How digital marketing works in real world. CO3. Email marketing and video and audio (Podcasting) marketing
SEMESTER VII	
CSC-401: Probability and Statistical Inference	CO1. Understand the concept of random variables and various discrete and continuous distributions. CO2. Apply probability distributions and central limit theorem to solve problems. CO3. Understand data properties using various statistical measures. CO4. Estimate population parameter (point and confidence interval) using statistical techniques.
CSC-402: Compiler Design	CO1. Know about different components of a compiler. CO2. Identify the similarities and differences among various parsing techniques. CO3. Learn Syntax directed definition, code optimization and code generation.
CSE-402: Digital Signal Processing	CO1. Interpret, represent and process discrete/digital signals and systems CO2. Thorough understanding of frequency domain analysis of discrete time signals. CO3. Ability to design & analyze DSP systems like FIR and IIR Filter etc. CO4. Practical implementation issues such as computational complexity, hardware resource limitations as well as cost of DSP systems or DSP Processors. CO5. Understand spectral analysis of the signals.

CSE-403: Data Mining and Warehousing	CO1. Understand data warehouse fundamentals and data mining principles. CO2. Design data warehouse with dimensional modeling and apply OLAP operations. CO3. Identify appropriate data mining algorithms to solve real world problems. CO4. Compare and evaluate different data mining techniques like clustering and association rule mining.
CSE-404: Fuzzy Logic	CO5. Difference between classical (crisp) set theory and fuzzy set theory. CO6. Learn different types of fuzzy membership functions. CO7. Apply fuzzy arithmetic to solve numerical problems. CO8. Discuss fuzzy reasoning and fuzzy clustering.
SEMESTER VIII	
CSC-451: Advanced Algorithms	CO1. Design and analyze programming problem statements. CO2. Choose appropriate data structures and algorithms, understand the ADT/libraries, and use it to design algorithms for a specific problem. CO3. Understand the necessary mathematical abstraction to solve problems. CO4. Come up with analysis of efficiency and proofs of correctness. CO5. Comprehend and select algorithm design approaches in a problem specific manner.
CSC-452: Information Theory and Coding	CO1. Understand the basics of information and coding theories. CO2. Discuss the various capacity reduction based coding techniques for text, audio and speech type of data. CO3. Illustrate various security oriented coding techniques for Block codes. CO4. Implement various error control techniques for Convolutional codes.
CSE-453: Machine Learning	CO1. Learn about supervised and unsupervised learning. CO2. Develop ML solutions to classification, regression and clustering problems. CO3. Learn about performance metrics to evaluate the models. CO4. Implement the models using MATLAB or Python.
CSE-454: Neural Networks	CO1. Know different types of neural networks. CO2. Identify the neural network algorithms, which are more appropriate for various types of learning tasks in various domains. CO3. Implement the NN models to solve real-world problems.

SEMESTER IX	
CSC-501: Parallel and Distributed Computing	CO1. Understand advancement in computer architecture in terms of performance, technology and latest trends of hardware. CO2. Explain ambiguity between parallel and distributed computing architecture. CO3. Appreciate the importance of parallel and distributed architecture for fast internet connectivity and linkage with cloud computing. CO4. Fundamental problems of distributed systems like clock synchronization, remote procedure call, group communication, etc., and techniques for solving these problems.
CSC-502: Combinatorics and Graph Theory	CO1. Know important classes of problems in graph theory. CO2. Formulate and prove fundamental theorems on graphs. CO3. In-depth knowledge of covering, connectivity and coloring problems in graphs.
CSE-504: Evolutionary Algorithms	CO1. Students will be able to learn genetic algorithm. CO2. Students will be able to learn PSO. CO3. Students will be able to learn about Differential evolution algorithm.
CSE-505: Software Defined Networks	CO1. Able to explain and build virtualized Software defined networks. CO2. Knowledge of the technology evolution leading to SDN as well as the Open Source role in SDN. CO3. Able to explore OpenFlow specifications to build Software defined networks. CO4. Acquire, investigate, analyze and synthesize information, concepts and theories relating to Network Management and SDN. CO5. Demonstrate knowledge software defined networking and its applications, including network programmability and virtualization.
SEMESTER X	
CSC-551: Research Methodology	CO1. Identify the research problem, hypothesis formation, formation of design and techniques. CO2. Perform Literature review. CO3. Write a research/review article.
CSC-552: Master's Thesis	CO1. Demonstrate a sound technical knowledge of their selected research topic. CO2. Undertake problem identification, formulation and solution. CO3. Design engineering solutions to the problem utilizing a systematic approach. CO4. Complete the research project resulting in a thesis publication.

Syllabus

(Semester I – Semester X)

SEMESTER I

CSC-101: Introduction to Computer Science Using Python **L | T | P (4 | 0 | 2)**

Computers and Programs: The Universal Machine, Program Power, what is Computer Science? Hardware Basics, Programming Languages, Problem Solving, Methods of Describing a Program, Flow Charts for Structured Programming, Computer Model, Procedures and Environments, Executing Procedure Calls and Returns, Global and Local Variables. Interfacing Procedures – Introduction – Reference Parameters – Automatic Protection of Arguments – Expression as Arguments in a Procedure Call – Function Procedures – Name Parameters – Parameters that Stand for Procedures and Functions – Recursion. Algorithm Design – Problem Solving Aspect – Top Down Design, Implementation of Algorithms – Fundamental Algorithms (Discuss the Design of Algorithms only). Program, Characteristics of a good program, Modular Approach, Programming style, Documentation and Program Maintenance, Compilers and Interpreters, Running and Debugging Programs, Syntax Errors, Run-Time Errors, Logical Errors, Concept of Structured Programming.

Inside a Python Program: Names, Expressions, Outputs, Assignment, Loops and Example Programs. Computing with Numbers and Strings: Numeric Data Types, The String Data Types, Input-Output as String Manipulation, File Processing, Objects and Graphics: Simple Graphics Programming, Using Graphical Objects, Choosing Coordinates, Interactive Graphics.

Defining Functions: Parameters, Processing, Return Values and Program Structure, Decision Structures: Simple Decisions, Two-Way Decisions, Multi-Way Decisions, Exception Handling, Loop Structures and Booleans.

Simulation and Design: Pseudorandom Numbers, Top-Down Design, Bottom-Up Implementation, Other Design Techniques, Defining Classes: Data Processing with Class, Objects and Encapsulation.

Data Collections: Applying Lists, Lists of Objects, designing with Lists and Classes, Case Study of Python Calculator, Non-Sequential Collection.

Textbooks:

1. John M. Zelle, "Python Programming: An Introduction to Computer Science", Franklin, Beedle & Associates Publishers, 3rd ed.
2. R.G. Dromey, "How to Solve It by Computer", Pearson Education, 5th ed., 2007.

References:

1. Richard L. Halterman, "Learning to Program with Python", Southern Adventist University Publisher, 2011.
2. C.H. Swaroop, "A Byte of Python", Ebshelf Inc., 2013.
3. Allen Downey, "Think Python: How to Think Like a Computer Scientist", Green Tea Press, 2012.

CSC-102: Digital System Design **L | T | P (4 | 0 | 2)**

Number Systems: Digital Computer, Number Systems– Number Representation, Binary, Octal, Hexadecimal, Unsigned and Signed Numbers, Arithmetic Operations, Fixed point and Floating Point representations, Use of different number systems in digital design, Binary Codes– BCD, EBCDIC, ASCII, Unicode, Gray codes, Excess-3, Error Detection and Correction codes.

Boolean Algebra and Digital Logic: Boolean Algebra– Definition, Boolean Logic and Theorems, Truth Tables, Logic Gates– AND, OR, NOT, NAND, NOR, XOR, Digital Circuit Characterization– Fan-in/Fan-out, Switching Functions, etc., Boolean Functions– Sum of Product (SoP) and Product of Sum (PoS) form, Karnaugh Maps, Simplification/Minimization of Boolean Functions using K-Map, K-Maps with Don't Care, Multiple Output Functions.

Combinational Logic & Circuit Design: Combinational Circuits– Analysis and Design Procedures, Circuits for Arithmetic Operations– Code Conversion, Binary Adder, Binary Subtractor, Decimal Adder, Magnitude Comparator, Decoders and Encoders, Multiplexers and Demultiplexers, Introduction to HDL– HDL Models of Combinational circuits, Introduction to VHDL and Basic VHDL Modelling.

Sequential Logic & Circuit Design: Sequential Elements– Latches and Flip Flops– Analysis and Design Procedures, Application of Flip Flops– Clock Generation, Counters, Registers, Shift Registers, State Machine Concepts– State Diagram, State Table, State Assignment and State Reduction/Minimization, HDL for Sequential Logic Circuits. Asynchronous Sequential Logic– Analysis and Design of Asynchronous Sequential Circuits, Reduction of State and Flow Tables, Race-free State Assignment, Hazards.

Memory & Programmable Logic Devices: Memory hierarchy, Memory technologies– Cache memory, Virtual memory, TLBs, Design of memory– ROM and RAM, Programmable Logic Array (PLA), Programmable Array Logic (PAL). Different Logic families– TTL, ECL, MOS, CMOS– operation, design and specification.

Input-Output Organization: Peripheral Devices, I/O Modules, Isolated vs. Memory-Mapped I/O, Asynchronous Data Transfer, Modes of Transfer– Programmed I/O, Interrupt-Driven I/O, Direct Memory Access (DMA) controller, I/O Processors (IOP).

Textbooks:

1. M. Morris Mano, Digital Logic and Computer Design, Pearson Education, 1st ed., 2004.
2. M. Morris Mano, Michael D. Ciletti, Digital Design: With an Introduction to the Verilog HDL, Pearson Education, 5th ed., 2014.

References:

1. David A. Patterson, John L. Hennessy, Computer Organization and Design: The Hardware/Software Interface, Morgan Kaufmann, 5th ed., 2016.
2. M. Morris Mano, Computer System Architecture, Pearson Education, 3rd ed., 2008.
3. John F. Wakerly, Digital Design Principles and Practices, Pearson Education, 4th ed., 2007.
4. Charles H. Roth Jr, Fundamentals of Logic Design, Jaico Publishing House, 5th ed., 2003.
5. Donald D. Givone, Digital Principles and Design, Tata McGraw Hill, 2003.

CSC-103: Basic Mathematics **L | T | P (3 | 1 | 0)**

Set Theory: Sets, Subsets, Equal Sets, Universal Sets, Finite and Infinite Sets, Operation on Sets– Union, Intersection and Complement, Cartesian Product, Relations, Equivalence Relation, Function– Domain and Range, injective, surjective and bijective functions, Inverse Functions, Trigonometric, Logarithmic and Exponential Functions.

Linear Algebra: Matrices, determinants, inverse, rank, solution of system of linear equations, types– symmetric, skew-symmetric, orthogonal, Hermitian, Skew-Hermitian and unitary matrices, eigenvalues and eigenvectors, Cayley-Hamilton Theorem and its applications, matrix diagonalization.

Differential Calculus: Function of single variable, limit, continuity, differentiability, application of derivatives, Rolle's Theorem, Mean value theorem, Taylor's and Maclaurin's expansions with remainders, Leibnitz Theorem, Intermediate forms, L'Hospital's rule, maxima and minima. Functions of two variables, limit, continuity, partial and total differentiation, Euler's theorem, maxima and minima of functions of two variables– Lagrange's method of multipliers, change of variables– Jacobians.

Ordinary Differential Equations: First order differential equations, separable equations, equations reducible to separable form, integrating factor, Bernoulli's equation. Second order linear differential equations with variable coefficients, method of variation of parameters, higher order linear differential equations with constant coefficients.

Integral Calculus: Integral as limit of sum, Fundamental theorem of integral calculus and mean value theorems, indefinite integrals, definite integrals, improper integrals, application of integrals, Beta and Gamma functions. Parametric equations and Polar coordinates.

Sequences and Series: Sequences, monotonic and bounded sequences, series, convergence and divergence of series, convergence test– Integral Test, Comparison Test, Alternating Series Test, Ratio Test, Root Test. Estimating value of an infinite series, Power series, Taylor series, Application of series, Binomial series.

Textbooks:

1. Erwin Kreyszig, "Advanced Engineering Mathematics", 9th ed., Wiley, 2011.
2. B.S. Grewal, "Elementary Engineering Mathematics", 43rd ed., Khanna Publishers, 2014.
3. H.K. Dass, "Advanced Engineering Mathematics", S. Chand & Company, 9th Revised Ed., 2001.
4. Shanti Narayan, P.K. Mittal, "Differential Calculus", 30th revised ed., S. Chand, 2005.
5. Shanti Narayan, P.K. Mittal, "Integral Calculus", 35th revised ed., S. Chand, 2005.
6. Sheldon Ross, "A First Course in Probability", 9th ed., Pearson Education, 2013.
7. Vijay K. Rohatgi, A.K. Md. Ehsanes Saleh, "An Introduction to Probability and Statistics", 2nd ed., Wiley, 2008.

EES-110: Environmental Studies **L | T | P (2 | 0 | 0)**

Introduction: The Multidisciplinary nature of Environmental Studies– Introduction, definition Objectives, Scope and Importance of Environmental Studies, Segments of Environment– Atmosphere, Hydrosphere, Lithosphere and Biosphere, Need for public awareness.

Resources: Natural Resources, Renewable and Nonrenewable resources, Forest resources, Water resources, Mineral resources, Food resources, Energy resources, Land resources. Role of an individual in conservation of natural resources.

Ecology & Ecosystem: Introduction and Definition, Structure/Components of Ecosystem, Types of Ecosystem, Functional attributes of an ecosystem, Productivity, Food chain relationships, Food Web, Ecological pyramids, Energy flow and Material Cycling.

Biodiversity & its Conservation: Introduction and Definition, Types of biodiversity, Biogeographical classification of India, Value of biodiversity, Hot spots of biodiversity, Threats to biodiversity, IUCN classification of species, Conservation of biodiversity-In-situ and Ex-situ conservation, Biosphere Reserves, National Parks, Wild life Sanctuaries, Zoological Gardens, Botanical Gardens, Seed Banks, Tissue Culture etc.

Pollution: Environmental Pollution– Pollutants, Types of pollutants, Effects of pollution on the environment, Types of environmental pollution– Air Pollution, Water Pollution, Soil Pollution, Noise Pollution, Thermal Pollution, Radioactive Pollution, Solid waste management (Definition, causes, effects and control of various pollution). Case studies. Disaster management– flood, earthquake, cyclone, landslides.

Social Issues and the Environment: Sustainable Development, Water Conservation and management, Rain water Harvesting, Climate change, Global warming, Acid Rain, Ozone layer depletion, Wastelands, wetland and their reclamation, Human population and the environment, Environmental laws. Case studies.

Textbooks:

1. S.K. Dhameja, “Environmental Studies”, Kataria & Sons, 2005.
2. A. Kaushik, C.P. Kaushik, “Basics of Environment and Ecology”, New Age International Publishers, 2010.
3. P. Singh, “Basics of Environment and Ecology”, New Age International Publishers, 2010.
4. A.K. De, “Environmental Chemistry”, New Age International Publishers, 2002.

CSS-100: Introduction to the IoT **L | T | P (1 | 0 | 1)**

Internet of Things (High level understanding): Definition, ideas, vision and examples, IoT devices vs. computers, Concepts and Technologies behind the IoT, Technological trends leading to IoT, Societal Impact of IoT– Benefits, Risks, Privacy and Security.

Internet of Things (Low level details): Embedded Systems– Meaning, Structure and Components, Interfacing and Interaction with the physical world– Sensors and Actuators.

Design and Implementation: Hardware and Software components of IoT devices– Integrated Circuits, Microcontrollers, IoT Operating System.

Networking and the Internet: Basics of networking, Internet Structure, Internet Protocols, MANETs.

Textbooks:

1. Arshdeep Bahga, Vijay Madisetti, Internet of Things: A Hands-On Approach, 1st ed., 2015.
2. Adrian McEwen, Hakim Cassimally, Designing the Internet of Things, Wiley, 2015.
3. Simon Monk, Programming Arduino: Getting Started with Sketches, McGraw Hill Education, 2nd ed., 2016.
4. Peter Waher, Learning Internet of Things, Packt Publishing, 2015.

CSG-101: Introduction to Computer Science Using Python **L | T | P (4 | 0 | 2)**

Computers and Programs: The Universal Machine, Program Power, what is Computer Science? Hardware Basics, Inside a Python Program: Names, Expressions, Outputs, Assignment, Loops and Example Programs. Computing with Numbers and Strings: Numeric Data Types, The String Data Types, Input-Output as String Manipulation.

Defining Functions: Parameters, Processing, Return Values and Program Structure, Decision Structures: Simple Decisions, Two-Way Decisions, Multi-Way Decisions, Exception Handling, Loop Structures and Booleans.

Simulation and Design: Pseudorandom Numbers, Top-Down Design, Bottom-Up Implementation, Other Design Techniques, Defining Classes: Data Processing with Class, Objects and Encapsulation.

Data Collections: Applying Lists, Lists of Objects, designing with Lists and Classes, Case Study of Python Calculator, Non-Sequential Collection.

Textbooks:

1. John M. Zelle, "Python Programming: An Introduction to Computer Science", Franklin, Beedle & Associates Publishers, 3rd ed.

References:

1. Richard L. Halterman, "Learning to Program with Python", Southern Adventist University Publisher, 2011.
2. C.H. Swaroop, "A Byte of Python", Ebshelf Inc., 2013.
3. Allen Downey, "Think Python: How to Think Like a Computer Scientist", Green Tea Press, 2012.

CSG-102: Digital System Design **L | T | P (5 | 1 | 0)**

Number Systems: Digital Computer, Number Systems– Number Representation, Binary, Octal, Hexadecimal, Unsigned and Signed Numbers, Arithmetic Operations, Fixed point and Floating Point representations, Use of different number systems in digital design, Binary Codes– BCD, EBCDIC, ASCII, Unicode, Gray codes, Excess-3, Error Detection and Correction codes.

Boolean Algebra and Digital Logic: Boolean Algebra– Definition, Boolean Logic and Theorems, Truth Tables, Logic Gates– AND, OR, NOT, NAND, NOR, XOR, Digital Circuit Characterization– Fan-in/Fan-out, Switching Functions, etc., Boolean Functions– Sum of Product (SoP) and Product of Sum (PoS) form, Karnaugh Maps, Simplification/Minimization of Boolean Functions using K-Map, K-Maps with Don't Care, Multiple Output Functions.

Combinational Logic & Circuit Design: Combinational Circuits– Analysis and Design Procedures, Circuits for Arithmetic Operations– Code Conversion, Binary Adder, Binary Subtractor, Decimal Adder, Magnitude Comparator, Decoders and Encoders, Multiplexers and Demultiplexers.

Sequential Logic & Circuit Design: Sequential Elements– Latches and Flip Flops– Analysis and Design Procedures, Application of Flip Flops– Clock Generation, Counters, Registers, Shift Registers, State Machine Concepts– State Diagram, State Table, State Assignment and State Reduction/Minimization. Asynchronous Sequential Logic– Analysis and Design of Asynchronous Sequential Circuits, Reduction of State and Flow Tables.

Memory & Programmable Logic Devices: Memory hierarchy, Memory technologies– Cache memory, Virtual memory, TLBs.

Textbooks:

1. M. Morris Mano, Digital Logic and Computer Design, Pearson Education, 1st ed., 2004.

References:

1. David A. Patterson, John L. Hennessy, Computer Organization and Design: The Hardware/Software Interface, Morgan Kaufmann, 5th ed., 2016.
2. M. Morris Mano, Computer System Architecture, Pearson Education, 3rd ed., 2008.
3. John F. Wakerly, Digital Design Principles and Practices, Pearson Education, 4th ed., 2007.
4. Charles H. Roth Jr, Fundamentals of Logic Design, Jaico Publishing House, Mumbai, 5th ed., 2003.
5. Donald D. Givone, Digital Principles and Design, Tata McGraw Hill, 2003.

CSG-103: Basic Electronics **L | T | P (5 | 1 | 0)**

Circuit Theory: History of electronics, network theorems, RC/RL circuits. RC/RL circuits.

Analog circuits: Diode, Half-wave and full-wave rectifiers, introduction to BJT. BJT equivalent circuit, amplifier analysis. op amp circuits, inverting and non-inverting amplifier, integrator, difference amp, instrumentation a. bode plots, op amp filters, non-idealities in op amp. triangle to sine converter, Schmitt triggers, applications. precision rectifiers, sinusoidal oscillators.

Digital circuits: Introduction, K maps, various combinatorial circuits. sequential circuits, flip-flops, counters. counters, design of synchronous counters. combination of counters, ADC, DAC**.**

Textbooks:

1. Ralph J. Smith, R.C. Dorf, Circuits, Devices and Systems, John Wiley, 1992.
2. Millman, Grabel, Microelectronics, McGraw Hill.
3. R.L. Boylestad, L. Nashelsky, Electronic Devices and Circuit Theory, Prentice Hall, 2002.
4. A. S. Sedra, K.C. Smith, Microelectronic circuits, Oxford University Press, 1998.
5. De Carlo, Lin, Linear circuit analysis, Oxford University Press, 2nd ed., 2001.

SEMESTER II

CSC-151: Programming in C **L | T | P (4 | 0 | 2)**

C Language Fundamentals: Computer program, Programming Languages– Low-Level & High-Level, compilation process, source code, object code, executable code. gcc/g++ compilers for Windows machine. Structure of a C Program, character set, keywords, identifiers, data types, constants, variables, expression, statement, comments. Input-Output in C, Operators & Expressions– relational operators, logical operators, assignment operators, increment/decrement operators, sizeof operator, bitwise operators, expressions and their evaluation, precedence and associativity, type conversion, typecast operator. Decision making constructs.

Functions: Modular programming– Advantages of using functions, Library functions, exploring C header files– stdio.h, conio.h, time.h, stdlib.h, process.h. User-defined functions– function declaration, definition and call, formal and actual arguments, scope and lifetime of variables, return statement, recursion, iterative vs. recursive way of problem solving, defining macros, macros without arguments, macros with arguments, macros vs. functions.

Arrays, Pointers and Strings: Arrays– declaration, initialization and processing one-dimensional and two-dimensional arrays, accessing array elements, array as function argument, linear and binary search, bubble sort, selection sort, insertion sort. Pointers– Memory, address operator, pointer variables, dereference operator, chain of pointer, pointer arithmetic, pointers and arrays, pointer to an array, array of pointers, pointers as function arguments, returning multiple values from a function, function returning pointers, pointers to functions, dynamic memory allocation, creation of one-dimensional and two-dimensional dynamic arrays, function pointers, memory leaks. Strings– string constant, string variable, reading/writing strings, format specifier for string, string manipulation using built-in and user-defined functions, array of strings- 2D array of characters, array of pointers to character.

Structure and Union: User-defined data type, defining a structure, union, declaration and initialization of structure variables, dot operator, copying and comparing structures, size of structure, array within structure, array of structures, nested structure, pointer within structure, pointer to structure, arrow operator, structure as function argument, function returning structure, self-referential structures, concept of linked list, bit fields, bit manipulation.

File management: Opening a file in various modes, concept of buffer, FILE structure, pointer to FILE, read/write operations on file, closing a file, random access to file, command line arguments.

Textbooks:

1. Brian W. Kernighan, Dennis M. Ritchie, "The C Programming Language", Prentice Hall, 2^{nd} ed., 1990.
2. Herbert Schildt, "C: The Complete Reference", McGraw Hill Education, 4^{th} ed., 2000.
3. Paul Deitel, Harvey Deitel, "C: How to program", Prentice Hall, 7^{th} ed., 2012.
4. Stephen Prata, "C Primer Plus", Sams Publishing, 5^{th} ed.
5. Yashavant Kanetkar, "Let Us C", BPB Publications, 13^{th} ed., 2013.

CSC-152: Computer Architecture **L | T | P (4 | 0 | 2)**

Register Transfer and Microoperations: Components of a computer system, Von Neumann architecture, Computer System Interconnection, Register Transfer Language, Register Transfer, Bus and Memory Transfers, Microoperations– Arithmetic, Logic and Shift.

Central Processing Unit: Computer Arithmetic– ALU, Integer Representation and Arithmetic, Floating-Point Representation and Arithmetic, Decimal Arithmetic, CU Implementation– Hardwired and Multi Programmed, Multiplier Control Unit, CPU Control Unit, Instruction Set Architecture– Addressing Modes and Design, CISC and RISC paradigm, Basic MIPS implementation– Building data path– Control Implementation scheme.

The 8086 microprocessor: Introduction to 8086 – Microprocessor architecture – Addressing modes, Instruction set and assembler directives- 8086 signals – Basic configurations – System bus timing –System design using 8086- System Bus Structure- Memory Interfacing, I/O interfacing, Parallel communication interface – Serial communication interface – D/A and A/D Interface.

Parallel Processing concepts: Instruction level parallelism, Parallel processing challenges, Flynn's classification, Pipelining, Vector Processing, Superscalar processors, Multi-core Processors– Multithreading, Multicore processor Architecture, Multiprocessor configurations – Coprocessor, closely coupled and loosely coupled configurations, Cache Coherence Protocols, Synchronization, Memory Consistency.

Textbooks:

1. M. Morris Mano, "Computer System Architecture", Pearson, 3rd ed., 2007.
2. Carl Hamacher, Zvonko Vranesic and Safwat Zaky, "Computer Organization", McGraw-Hill, 5th ed., 2002.
3. Yu-Cheng Liu, Glenn A. Gibson, "Microcomputer Systems: The 8086 / 8088 Family: Architecture, Programming and Design", Prentice Hall of India, 2nd ed., 2007.

References:

1. David A. Patterson, John L. Hennessy, "Computer Organization and Design: The Hardware/Software Interface", Morgan Kaufmann, 3rd ed., 2007.
2. John L. Hennessy, David A. Patterson, "Computer Architecture: A Quantitative Approach", Morgan Kaufmann, 5th ed., 2012.
3. John P. Hayes, "Computer Architecture and Organization", McGraw-Hill Education, 2nd ed., 1998.
4. William Stallings, "Computer Organization and Architecture – Designing for Performance", Pearson Education, 6th ed., 2003.
5. Doughlas V. Hall, "Microprocessors and Interfacing, Programming and Hardware", TMH, 2012.

Computer Architecture Lab

Introduction to Assembly Language Programming: The Need and Use of the Assembly Language, Assembly Program Execution, An Assembly Program and its Components, The Program Annotation, Directives, Input Output in Assembly Program, Interrupts, DOS Function Calls (Using INT 21H), The Types of Assembly Programs, COM Programs, EXE Programs, How to Write Good Assembly Programs.

Assembly Language Programming (Part-I): Simple Assembly Programs, Data Transfer, Simple Arithmetic Application, Application Using Shift Operations, Larger of the Two Numbers, Programming with Loops and Comparisons, Simple Program Loops, Find the Largest and the Smallest Array Values, Character Coded Data, Code Conversion, Programming for Arithmetic and String Operations, String Processing, Some More Arithmetic Problems.

Assembly Language Programming (Part-II): Use of Arrays in Assembly, Modular Programming, the stack, FAR and NEAR Procedures, Parameter Passing in Procedures, External Procedures, Interfacing Assembly Language Routines to High Level Language, Programs, Simple Interfacing, Interfacing Subroutines with Parameter Passing, Interrupts, Device Drivers in Assembly.

CSC-153: Discrete Mathematics **L | T | P (3 | 1 | 0)**

Logic and Proofs: Propositional Logic– Binary logic and propositions, Propositional Variables, Truth table, Logical connectives– Negation, Conjunction, Disjunction, Conditional, Biconditional, Universal connectives, Well-formed Formulas, Tautology, Contradiction and Contingency, Propositional Equivalences, Duality, Predicate Logic– Predicates, Quantifiers– Existential and Universal quantifier, Predicate formulas, Equivalence of formulas involving quantifiers, Normal forms– CNF/DNF, PCNF/PDNF, Normal forms for First Order Logic– Prenex Normal Form, Rules of Inference.

Proof Techniques– Introduction to Proof, Definitions– Theorem, Lemma, Corollary and Conjecture, Methods of Proof– Direct Proofs, Indirect Proofs– Proof by Contraposition, Proof by Contradiction, Proof by Cases, Mathematical Induction.

Counting and Recurrence: Set Theory, Countable and Countably Infinite Sets, Pigeonhole Principle, Permutation and Combination, Principle of Inclusion and Exclusion, Generating functions– Definition, Generating Permutations and Combinations. Recurrence– Recurrence Relations, Linear Recurrence Relations with constant coefficients and their solution, Solving Linear Recurrence Relations using Generating Functions.

Binary and Ordered Relations: Binary Relation, Properties of Binary Relations– Reflexive, Symmetric and Transitive Relation, Equivalence Relation, Closure of Relations– Reflexive, Symmetric and Transitive Closure, Warshall's algorithm, Ordered Relation– Partial Order and Posets, Hasse diagram of Poset, Maximal, Minimal, Maximum and Minimum of poset, glb and lub, Isomorphic ordered set, Well ordered set, Lattice, Properties of lattice, Distributed and Complemented lattice, Applications of Lattice, Topological Sort.

Graph Theory: Definition of Graph, Types– Directed and Undirected Graph, Complete Graph, Bipartite Graph, Multigraph, Weighted Graph, Graph Representation– Adjacency matrix and Adjacency list, Graph Isomorphism, Connectivity and Path, Euler and Hamiltonian Paths and Circuits, Shortest path– Dijkstra's algorithm, Planar Graph, Euler's theorem for Planar Graphs, Graph Coloring.

Trees– Basic terminology and properties, Tree Traversal– Inorder, Preorder and Postorder, Expression Trees– Infix, Prefix and Postfix notations, Spanning Trees– Kruskal's and Prim's algorithms for Minimum Spanning Trees (MST).

Algebra: Definition and elementary properties of Semigroups, Monoids, Groups, Subgroups, Generators and Cyclic group, Permutation group, Cosets, Lagrange's Theorem, Rings, Integral Domains and Fields.

Textbooks:

1. Kenneth H. Rosen, "Discrete Mathematics and Its Applications", Tata McGraw Hill, 7^{th} ed., 2012.
2. C. L. Liu, "Elements of Discrete Mathematics", McGraw Hill, 2^{nd} ed., 1986.
3. Bernard Kolman, Robert C. Busby, Sharon Cutler Ross, "Discrete Mathematical Structures", Pearson Education, 6^{th} ed., 2008.
4. J. P. Tremblay, R. Manohar, "Discrete Mathematical Structures with Applications to Computer Science", Tata McGraw Hill, 1^{st} ed., 2001.
5. Susanna S. Epp, "Discrete Mathematics with Applications", 4^{th} ed., 2010.

CSC-154: Project-1: Embedded System Development **(2 credits)**

The focus of this project lies in designing a system/device for a real-world application. The skills required to complete this project are basic knowledge of IoT and Embedded System with some C/Python programming. All of the requirements are covered in the first year of the curriculum.

ENG-151: English **L | T | P (2 | 0 | 0)**

Grammar: Sentence structure, Noun, verb, adjective, pronoun, adverb, preposition, conjunction, interjection. Subject and object, Subject verb agreement (Introduction), Tenses (Introduction).

Formal Letter, Comprehension and interpretation of given lines, Precise or summary of a given article, Phonetic Transcription (Pronunciation). Use of stress, pause, intonation, voice modulation while reading and speaking. Use of stress, pause, intonation, modulation during reading and verbal communication.

Forms of communication-verbal, non-verbal and written. Essay. (Art of Essay writing as an expression of the personality of the writer) Aim of an essay; as used in interviews, open exams and admission to international universities.

Common errors in English Language. Group Discussion. Formal Letter. Communication through presentations PPT.

Textbooks:

1. Raymond Murphy, "English Grammar in Use", Cambridge University Press, 4th ed., 2012.
2. Martin Hewings, "Advanced English Grammar", Cambridge University Press, 4th ed., 1999.
3. Meenakshi Raman, Sangeeta Sharma, "Technical Communication: Principles and Practice", Oxford, 2nd ed., 2011.

CSS-150: Unix System Administration and Shell Programming **L | T | P (0 | 0 | 2)**

Getting Started: File Management, Directories, File Permission, Environment, Basic Utilities, Pipes & Filters, Processes, Communication, The vi Editor.

Shell Programming: What is Shell? What is shell script, Writing and executing the shell script, Shell variable (user defined and system variables), System calls, Using system calls, Pipes and Filters, Decision making in Shell Scripts (If else, switch), Loops in shell, Functions, Utility programs (cut, paste, join, tr, uniq utilities), Pattern matching utility (grep), Special Variables, Using Arrays, Basic Operators, Shell Substitutions, Quoting Mechanisms, IO Redirections, Shell Functions, Manpage Help, Simple filters, the process, TCP/IP Networking Tools, The X Window System, Filters using regular expressions.

Customizing the environment, Programming in C/C++ in UNIX environment, Shell programming, Perl, System and network administration**.**

Advanced Unix: Regular Expressions, File System Basics, User Administration, System Performance, System Logging, Signals and Traps.

Textbooks:

1. Sumitabha, Das, "Unix Concepts and Applications", Tata McGraw-Hill Education, 2006.
2. Evi Nemeth, Garth Synder, Trent R. Hein, Ben Whaley, "Unix and Linux System Administration Handbook", Pearson Education, 2nd ed., 2010.
3. Venkatesh Murthy, "Introduction to Unix & Shell Programming", Pearson Education, 2006.
4. Stephan Prata, "Advanced Unix – A Programmers Guide", BPB PUB.
5. Brian W. Kernighan, Rob Pike, "The Unix Programming Environment", PHI.
6. W. Richard Stevens, Stephen A. Rago, "Advanced Programming in the UNIX Environment", Pearson Education, 2nd ed., 2011.
7. The UNIX System Manuals.

CSG-151: Programming in C **L | T | P (4 | 0 | 2)**

C Language Fundamentals: Computer program, Programming Languages– Low-Level & High-Level, compilation process, source code, object code, executable code, Basics of computer memory architecture. gcc/g++ compilers for Windows machine, IDEs like Dev-cpp, Code Blocks. Structure of a C Program, character set, keywords, identifiers, data types, constants, variables, expression, statement, comments. Input-Output in C– reading/writing a character, formatted I/O for int, float and char data types. Operators & Expressions– arithmetic operators, integer arithmetic, floating-point arithmetic, mixed mode arithmetic, relational operators, logical operators, assignment operators, increment/decrement operators, sizeof operator, bitwise operators, expressions and their evaluation, precedence and associativity, type conversion, implicit & explicit, typecast operator. Branching & Looping– if, if...else, else if, switch-case, loops, for, while, do...while, break, continue.

Functions: Modular programming– Advantages of using functions, Library functions. User-defined functions– function declaration, definition and call, formal and actual arguments, scope and lifetime of variables, return statement, recursion, iterative vs. recursive way of problem solving.

Arrays, Pointers and Strings: Arrays– declaration, initialization and processing one-dimensional and two-dimensional arrays, accessing array elements, array as function argument, linear and binary search, bubble sort, selection sort, insertion sort. Pointers– Memory, address operator, declaration of pointer variables, assigning address to pointer variable, dereference operator, pointer arithmetic, pointers and arrays, pointer to an array, array of pointers, pointers as function arguments, returning multiple values from a function, dynamic memory allocation, creation of one-dimensional dynamic arrays. Strings– string constant, string variable, reading/writing strings, string manipulation using built-in functions, array of strings- 2D array of characters.

Structure and Union: User-defined data type, defining a structure, union, declaration and initialization of structure variables, dot operator, copying and comparing structures, size of structure, array within structure, array of structures, nested structure, arrow operator, structure as function argument, function returning structure.

File management: Opening a file in various modes, concept of buffer, FILE structure, pointer to FILE, read/write operations on file, closing a file, random access to file.

Textbooks:

1. Herbert Schildt, "C: The Complete Reference", McGraw Hill Education, 4^{th} ed., 2000.
2. Paul Deitel, Harvey Deitel, "C: How to program", Prentice Hall, 7^{th} ed., 2012.
3. Stephen Prata, "C Primer Plus", Sams Publishing, 5^{th} ed.
4. Yashavant Kanetkar, "Let Us C", BPB Publications, 13^{th} ed., 2013.

<u>CSG-152: Computer Architecture</u> **L | T | P (5 | 1 | 0)**

Register Transfer and Microoperations: Computer Architecture vs. Organization, Von Neumann architecture, Computer System Interconnection, Register Transfer Language, Register Transfer, Bus and Memory Transfers, Microoperations– Arithmetic, Logic and Shift.

Central Processing Unit: Computer Arithmetic– ALU, Integer Representation and Arithmetic, Floating-Point Representation and Arithmetic, Decimal Arithmetic, CU Implementation– Hardwired and Multi Programmed, Instruction Set Architecture– Addressing Modes and Design, CISC and RISC paradigm, Parallel Processing concepts, Pipelining, Vector Processing, Superscalar processors, Multi-core Processors– Multithreading, Multicore processor Architecture, Cache Coherence Protocols, Synchronization, Memory Consistency.

Input-Output Organization: Peripheral Devices, I/O Modules, Isolated vs. Memory-Mapped I/O, Asynchronous Data Transfer, Modes of Transfer– Programmed I/O, Interrupt-Driven I/O, Direct Memory Access (DMA) controller, I/O Processors (IOP).

Memory Organization: Memory System and Hierarchy, Semiconductor Memory concepts, CPU-memory interaction, Cache Memory, Virtual Memory, Main Memory, DRAM and SRAM, External Memory, Associative Memory.

Text Books:

1. M. Morris Mano, “Computer System Architecture”, Pearson, 3rd ed., 2007.

References:

1. John P. Hayes, “Computer Architecture and Organization”, McGraw-Hill Education, 2nd ed., 1998.
2. William Stallings, “Computer Organization and Architecture – Designing for Performance”, Pearson Education, 6th ed., 2003.

CSG-153: Embedded System Design **L | T | P (5 | 1 | 0)**

Introduction: Processors– General Purpose and ASIPs Processor, Designing a single purpose processor, optimization issues. FPGA, Behavior Synthesis on FPGAs using VHDL.

Sensors and Signals: Discretization of Signals, A/D converter, Quantization Noise, D/A Converter, Arduino Uno, Serial Communication and Timer, Controller Design using Arduino.

Embedded Systems: Power Aware Embedded System, SD and DD Algorithm, Parallel operations and VLIW, DSP Application and Address Generation Unit, Real Time OS, RMS Algorithm, EDF Algorithm, Priority Inversion and Priority Inheritance Protocol, Modeling and Specification, FSM and Statecharts, Program State Machines, SDL, Data Flow Model, Hardware Synthesis, Digital Camera Design, HW-SW Partitioning, Optimization, Simulation, Formal Verification.

Textbooks:

1. Arnold S. Berger, "Embedded Systems Design", CRC Press, 2001.
2. Michael Barr, "Programming Embedded Systems in C and C++", O'Reilly, 1999.

SEMESTER III

CSC-201: Object Oriented Programming Using C++ **L | T | P (4 | 0 | 2)**

OOPS paradigm: Object & Classes, Links and Associations, Generalization and Inheritance, Aggregation, Abstract Classes, A sample Object Model, Multiple Inheritance, Meta Data, Candidate Keys, Constraints.

Dynamic Modeling: Event and States, Operations and Methods, Nested State, Diagrams, Concurrency, Relation of Object and Dynamic Models, Advanced Dynamic Model Concepts Keys, Constraints. Functional Modeling: Functional Models, Data Flow Diagrams, Specifying Operations, Constraints, a Sample Functional Model.

Programming in C++: Classes and Objects in C++, Functions, Constructors, Inheritance, Function Overloading, Operator Overloading, I/O Operations, Real Life Applications, Extended Classes Pointer, Virtual Functions, Polymorphisms, Class Templates, Function Templates, Exception Handling, C++ I/O.

Translating Object Oriented Design into an Implementation, OMT Methodologies, Examples and Case Studies to Demonstrate Methodology, Comparison of Methodology, SA/SD and JSD.

Introduction to Standard Template Library: STL– containers, algorithms, iterators, other STL elements. Container classes– vector, list, deque, set, multiset, map, multimap, stack, queue, priority queue, bitset. Function objects– unary and binary function objects, built-in function objects, create a function object. The string class– introduction, manipulating string objects, relational operations, comparison.

Textbooks:

1. James R. Rambaugh, "Object Oriented Design and Modeling", PHI.
2. Booch Grady, "Object Oriented Analysis and Design with Application", Pearson, 3rd ed.
3. Dillon and Lee, "Object Oriented Conceptual Modeling", New Delhi PHI-1993.
4. Stephen R. Shah, "Introduction to Object Oriented Analysis and Design", TMH.
5. Berzin Joseph, "Data Abstraction; The Object Oriented Approach Using C++", McGraw Hill.
6. Herbert Schildt, "C++: The Complete Reference", McGraw Hill, 4th ed., 2003.
7. Walter Savitch, "Absolute C++", Pearson, 5th ed., 2012.
8. Lipman, Stanley B, Jonsce Lajole, "C++ Primer Reading", AWL-1999
9. Bjarne Stroustrup, "The C++ Programming Language", Pearson, 3rd ed., 2002.
10. E. Balagurusamy, "Object Oriented Programming with C++", TMH, 6th ed., 2013.

CSC-202: Operating Systems **L | T | P (4 | 0 | 2)**

Operating System Overview: Operating Systems– objectives and functions, evolution– early Operating Systems, Parallel systems, Distributed Systems, Process Control & Real-time Systems. Modern Operating Systems, Virtual Machines, OS Design considerations for Multiprocessor and Multicore architectures. OS Organization– Processor and user modes, Kernel, System Calls, System Programs, System Boot, Overview and Booting process of various OS– Microsoft Windows, Modern UNIX, Linux, Android.

Process Management: System view of the process and resources, process abstraction, process hierarchy, process control, execution of the OS. Threads– concept, issues, libraries, thread programming using pthread, multicore processors and threads, multithreading models, process and thread management in Linux, Android and Windows. Process scheduling– Uniprocessor, Multiprocessor and Real-time scheduling algorithms, Traditional UNIX scheduling, Linux scheduling.
Concurrency– Process/Thread synchronization, Mutual Exclusion Principles of Concurrency, Critical Section Problem, Hardware support, OS support (semaphores, mutex), Programming Language support (monitors), Classical synchronization problems– Readers/Writers problem, Producer/Consumer problem.
Deadlocks– Deadlock characterization, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Dining Philosophers Problem. Linux and Android inter process communication (IPC) and concurrency mechanisms.

Memory Management: Logical vs. Physical Address space, Memory Partitioning– Fixed and Dynamic Partitioning, Buddy System, Relocation, Paging, Segmentation. Virtual Memory– Demand Paging, Page Replacement, Frames allocation, Thrashing, Allocating Kernel Memory. Memory Management in Linux.

I/O and File Management: I/O Devices, Buffering, Disk Scheduling, Sector Queuing, Linux I/O. File– File Concept, File Organization, Access Methods, File Sharing and Protection, Logical and Physical File System, Directory Structure, Allocation Methods– Contiguous, Sequential and Indexed Allocation, Linux Virtual File System.

Case Study: Linux and Windows Operating Systems.

Textbooks:

1. Abraham Silberschatz, Peter B. Galvin, Greg Gagne, "Operating System Concepts", John Wiley Publications, 8th ed., 2008.
2. William Stallings, "Operating Systems: Internals and Design Principles", Pearson, 7th ed., 2013.
3. Robert Love, "Linux Kernel Development", Pearson, 1st ed., 2010.

References:

1. Dhananjay M. Dhamdhere, "Systems Programming and Operating Systems", Tata McGraw-Hill, 2nd ed., 1999.
2. Gary Nutt, "Operating Systems: A Modern Perspective", Pearson, 3rd ed., 2009.
3. Maurice J. Bach, "The Design of the UNIX Operating System", PHI.

CSC-203: Data Structures **L | T | P (4 | 0 | 2)**

Introduction to problem solving approach: Algorithmic solution, analysis of algorithms– space and time complexity, asymptotic analysis, Big-O, Θ and Ω notations, step counting and time complexity analysis using C language, data structures and algorithms, linear and nonlinear data structures.

Sequential Lists and Linked Lists: Sequential lists, arrays– single and multi-dimensional arrays, sparse matrix, algorithm to store sparse matrices, singly, doubly and circular linked lists, list traversal algorithms, stacks– array implementation and linked list implementation, applications of stack, queues– array implementation and linked list implementation, circular queue and dequeue.

Searching and Sorting: Searching algorithms– linear search, binary search, comparison of linear and binary search, constant time search using hashing, hash functions, collision resolution techniques– linear probing and chaining, Sorting algorithms– bubble sort, selection sort, insertion sort, merge sort, quick sort, radix sort, shell sort, bucket sort, comparison of sorting techniques.

Trees: Basic terminology, binary tree, depth, height and number of nodes, tree traversals, expression trees, post/pre/infix notation, application of binary tree in Huffman coding, binary search tree, search, insertion and deletion operations in BST, balanced BST, AVL tree, insertion and deletion in AVL tree, overview of m-way search tree and B tree, priority queues, binary heap, heapsort.

Graphs: Graph theory terminology, graph representation– adjacency matrix and adjacency list, graph traversal algorithms– Breadth First Search and connected components, Depth First Search and strongly connected components, applications of BFS and DFS, Minimum Spanning Trees, Prim's and Kruskal's algorithm, Union-find data structure, Single source shortest path, Dijkstra's algorithm for shortest path, Floyd Warshall's algorithm for All-pairs shortest path.

Textbooks:

1. Yedidyah Langsam, Moshe J. Augenstein, Aaron M. Tenenbaum, "Data Structures using C and C++", 2^{nd} ed., Pearson Education, 2006.
2. Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, "Fundamentals of Data Structures in C", Universities press, 2^{nd} ed., 2008.
3. Robert Sedgewick, Kevin Wayne, "Algorithms", Pearson Education, 4^{th} ed., 2011.
4. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", PHI, 3^{rd} ed., 2010.
5. Seymour Lipschutz, "Data Structures", Schaum's outlines, McGraw Hill Education, 1^{st} ed., 2014.
6. Donald E. Knuth, "The Art of Computer Programming", Vol. 1 and Vol. 3.

CSS-200: Client-side Web Technologies **L | T | P (0 | 0 | 2)**

HTML and CSS: Concept of WWW, Internet and WWW, HTTP Protocol: Request and Response, Web browser and Web servers, Features of Web 2.0, Basic HTML, formatting and fonts, commenting code, color, hyperlink, lists, tables, images, simple HTML forms, website structure.

CSS: Need for CSS, introduction to CSS, basic syntax and structure, using CSS, Inline styles, embedded style sheets, conflicting styles, linking external style sheets, positioning elements, backgrounds, element dimensions, background images, colors and properties, manipulating texts, using fonts, borders and boxes, margins, padding lists, positioning using CSS, CSS2, Overview and features of CSS3.

Javascript: Client side scripting with JavaScript, variables, functions, conditions, loops and repetition, Pop up boxes, Advance JavaScript: Javascript and objects, JavaScript own objects, the DOM and web browser environments, Manipulation using DOM, forms and validations, DHTML: Combining HTML, CSS and Javascript, Events and buttons.

XML: XML, Uses of XML, well-formed XML, XML Schema– Built-in data types, Simple and Complex types, Elements, Attributes, Unions and Lists, Parsing XML, Tags, text, comments and empty elements. XML Declaration, Processing Instructions, Errors in XML.

XML Namespaces: Need for namespaces, How XML namespaces work, URIs, when to use namespace. Document Type Definition (DTD), External and Internal DTDs, sharing vocabularies, Anatomy of DTD, Developing DTDs, DTD Limitations, Overview of the XML DOM, SAX. XPath– XPath Expressions, Operators and Special Characters, Filters, Location Paths, XPath functions, XSL– XSL versions, XSL processor, XSL numbers, XSL functions, XSLT, XLink, XPointer.

AJAX: Evolution of AJAX, AJAX Framework, Web applications with AJAX, AJAX with PHP, AJAX with Databases.

References:

1. Chuckmusiano and Bill Kenndy, HTML The Definite Guide, O Reilly, 2000.
2. Joseph Schmuller, Dynamic HTML, BPB publications, 2000.
3. Jeffrey C Jackson, Web Technology– A Computer Science perspective, Pearson Education, 2007.
4. Dave Taylor, “Creating Cool Web Sites with HTML, XHTML, and CSS”, Wiley.
5. Virginia DeBolt, “Integrated HTML and CSS: A Smarter, Faster Way to Learn”, Wiley, 2008.
6. Patrick Carey, “New Perspectives on HTML, XHTML and XML”, 3rd ed., Course Tech.
7. Michael Young, “XML Step by Step”, 2nd ed., Microsoft Press, 2002.

CSG-201: Object Oriented Programming Using C++ L | T | P (4 | 0 | 2)

OOPS paradigm: Object & Classes, Links and Associations, Generalization and Inheritance, Aggregation, Abstract Classes, A sample Object Model, Multiple Inheritance, Meta Data, Candidate Keys, Constraints.

Dynamic Modeling: Event and States, Operations and Methods, Nested State, Diagrams, Concurrency, Relation of Object and Dynamic Models, Advanced Dynamic Model Concepts Keys, Constraints. Functional Modeling: Functional Models, Data Flow Diagrams, Specifying Operations, Constraints, a Sample Functional Model.

Programming in C++: Classes and Objects in C++, Functions, Constructors, Inheritance, Function Overloading, Operator Overloading, I/O Operations, Real Life Applications, Extended Classes Pointer, Virtual Functions, Polymorphisms, Class Templates, Function Templates, Exception Handling, C++ I/O.

Textbooks:

1. James R. Rambaugh, "Object Oriented Design and Modeling", PHI.
2. Booch Grady, "Object Oriented Analysis and Design with Application", Pearson, 3rd ed.
3. Dillon and Lee, "Object Oriented Conceptual Modeling", New Delhi PHI-1993.
4. Stephen R.Shah, " Introduction to Object Oriented Analysis and Design", TMH.
5. Berzin Joseph, "Data Abstraction; The Object Oriented Approach Using C++", McGraw Hill.
6. Herbert Schildt, "C++: The Complete Reference", McGraw Hill, 4th ed., 2003.
7. Walter Savitch, "Absolute C++", Pearson, 5th ed., 2012.
8. Lipman, Stanley B, Jonsce Lajole, "C++ Primer Reading", AWL-1999
9. Bjarne Stroustrup, "The C++ Programming Language", Pearson, 3rd ed., 2002.
10. E. Balagurusamy, "Object Oriented Programming with C++", TMH, 6th ed., 2013.

CSG-202: Operating Systems **L | T | P (5 | 1 | 0)**

Operating System Overview: Operating Systems– objectives and functions, evolution–early Operating Systems, Parallel systems, Distributed Systems, Process Control & Real-time Systems. Modern Operating Systems, Virtual Machines, OS Design considerations for Multiprocessor and Multicore architectures. OS Organization– Processor and user modes, Kernel, System Calls, System Programs, System Boot, Overview and Booting process of various OS– Microsoft Windows, Modern UNIX, Linux, Android.

Process Management: System view of the process and resources, process abstraction, process hierarchy, process control, execution of the OS. Threads– concept, issues, libraries, thread programming using pthread, multicore processors and threads, multithreading models, process and thread management in Linux, Android and Windows. Process scheduling– Uniprocessor, Multiprocessor and Real-time scheduling algorithms.
Concurrency– Process/Thread synchronization, Mutual Exclusion Principles of Concurrency, Critical Section Problem, Hardware support, OS support (semaphores, mutex), Programming Language support (monitors), Classical synchronization problems– Readers/Writers problem, Producer/Consumer problem.
Deadlocks– Deadlock characterization, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Dining Philosophers Problem.

Memory Management: Logical vs. Physical Address space, Memory Partitioning– Fixed and Dynamic Partitioning, Buddy System, Relocation, Paging, Segmentation. Virtual Memory– Demand Paging, Page Replacement, Frames allocation, Thrashing.

I/O and File Management: I/O Devices, Buffering, Disk Scheduling. File– File Concept, File Organization, Access Methods, File Sharing and Protection, Logical and Physical File System, Directory Structure, Allocation Methods– Contiguous, Sequential and Indexed Allocation.

Case Study: Linux and Windows Operating Systems.

Textbooks:

1. Abraham Silberschatz, Peter B. Galvin, Greg Gagne, "Operating System Concepts", John Wiley Publications, 8th ed., 2008.
2. William Stallings, "Operating Systems: Internals and Design Principles", Pearson, 7th ed., 2013.

References:

1. Dhananjay M. Dhamdhere, "Systems Programming and Operating Systems", Tata McGraw-Hill, 2nd ed., 1999.
2. Gary Nutt, "Operating Systems: A Modern Perspective", Pearson, 3rd ed., 2009.

CSG-203: Econometrics **L | T | P (5 | 1 | 0)**

Introduction to Econometrics, Simple linear regression analysis, Multiple linear regression analysis, Predictions in linear regression model, Generalized and weighted least squares estimation, Regression analysis under linear restrictions, Multicollinearity, Heteroscedasticity, Autocorrelation, Dummy variables models, Specification error analysis, Tests for structural change and stability, Asymptotic theory and stochastic regressors, Stein-rule estimation, Instrumental variable estimation, Measurement error models, Simultaneous equations models, Seemingly unrelated regression equations models.

References:

1. Jack Johnston and John DiNardo (1997): Econometric Methods, 3rd ed., McGraw Hill.
2. G.S. Maddala (2002): Introduction to Econometrics, 3rd ed., John Wiley.
3. Damodar Gujrati (2003): Basic Econometrics, McGraw Hill.
4. William Greene (2008): Econometric Analysis, Prentice Hall, 6th ed.
5. J.M. Woolridge (2002): Introductory Econometrics- A Modern Approach, South-Western College.

SEMESTER IV

CSC-251: Programming in Java **L | T | P (4 | 0 |2)**

Introduction to Java: Java Architecture and Features, Differences between C++ and Java, Compiling and Executing a Java Program, Variables, Constants, Keywords, Data Types, Operators, Expressions, Comments, Using Classes, Controlling Access to Class Members, Basic Program, Decision Making Constructs.

Java Methods, Arrays, Strings and I/O: Java, Class Constructors, Method Overloading, final classes, Object class, Garbage Collection, Arrays, Java Strings: The Java String class, String Buffer Classes. Simple I/O using System.out and the Scanner class, Byte and Character streams, Reading/Writing from console and files.

Inheritance, Interfaces, Packages, Enumerations, Autoboxing and Metadata: Single Level and Multilevel Inheritance, Method Overriding, Dynamic Method Dispatch, Abstract Classes, Interfaces and Packages, Package and Class Visibility, Standard Java Packages (util, lang, io, net), Wrapper Classes, Autoboxing/Unboxing, Enumerations and Metadata.

Exception Handling, Threading, Networking and Database Connectivity: Exception types, uncaught exceptions, throw, built-in exceptions, creating your own exceptions; Multi-threading: The Thread class and Runnable interface. Thread prioritization, synchronization and communication, suspending/resuming threads. Using java.net package, Overview of TCP/IP and Datagram programming. Accessing and manipulating databases using JDBC.

Applets and Event Handling: Java Applets: Introduction to Applets, Writing Java Applets, working with Graphics, Incorporating Images & Sounds. Event Handling Mechanisms, Listener Interfaces, Adapter and Inner Classes. The design and Implementation of GUIs using the AWT controls, Swing components of Java Foundation Classes such as labels, buttons, textfields, layout managers, menus, events and listeners; Graphic objects for drawing figures such as lines, rectangles, ovals, using different fonts.

Advanced Topics: Java Beans and Web Servers, BDK, Introduction to EJB, Java Beans API, Introduction to Servlets, Lifecycle, JSDK, Servlet API, Servlet Packages: HTTP package, Working with Http request and response, Security Issues. Java Script: Data types, variables, operators, conditional statements, array object, date object, string object, Dynamic Positioning and front end validation. JSP, J2SE.

Textbooks:

1. Herbert Schildt, “Java, The Complete Reference”, TMH, 7th ed., 2007.
2. Ken Arnold, James Gosling, David Homes, "The Java Programming Language", 4th ed., 2005.
3. Cay S. Horstmann, GaryCornell, "Core Java 2 Volume 1 and 2”, Prentice Hall, 9th ed., 2012.
4. Bruce Eckel, "Thinking in Java", PHI, 3rd ed., 2002.
5. Paul Deitel, Harvey Deitel, "Java: How to Program", Prentice Hall, 10th ed., 2011.
6. Bert Bates and Kathy Sierra "Head First Java", Orielly Media Inc., 2nd ed., 2005.
7. Elliotte Rusty Harold, “Java Network Programming”, O’Reilly publishers, 2000
8. Ed Roman, “Mastering Enterprise Java Beans”, John Wiley & Sons Inc., 1999.

CSC-252: Database Systems **L | T | P (4 | 0 | 2)**

Introduction: Overview of databases, Data models, DBMS architecture and data independence, History of Database Systems. Entity-Relationship Modeling: Basic concepts, constraints, keys, Design issues, weak entities, enhanced E-R, Sub Classes, Super classes, inheritance, specialization and generalization.

Relational Data Model and Normalization: Relational model concepts, relational constraints, relational algebra, relational calculus. SQL: basic queries, nested subqueries, aggregate functions, null values, complex queries, database modification commands, programming using SQL, embedded SQL, dynamic SQL. Database Design– Functional dependencies, Normalization, Normal form– 1NF, 2NF, 3NF, BCNF.

Integrity and Security: Domain Constraints, Referential Integrity Constraints, Assertions, Triggers, Security and Authorization– Authorization in SQL, Encryption and Authentication.

File Organization: Indexed sequential access files, implementation using B+ trees, hashing, hashing functions, collision resolution, extendible hashing, dynamic hashing approach implementation and performance.

Transaction and Concurrency Control: Transaction concept, Transaction state, ACID properties and their implementation. Concurrency Control– Lock Based Protocols, Timestamp Based Protocols, Validation Based Protocols, Multiple Granularity. Recovery System– Failure Classification, Storage Structure, Recovery and Atomicity, Log based recovery.

Textbooks:

1. Ramez Elmasri, Shamkant B. Navathe, "Fundamentals of Database Systems", Pearson Education, 5th ed., 2008.
2. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, "Database Concepts", McGraw-Hill, 6th ed., 2013.
3. R. Ramakrishanan, J. Gehrke, "Database Management Systems" McGraw-Hill, 3rd ed., 2002.
4. Peter Rob, Carlos Coronel, "Database Systems: Design, Implementation and Management", 7th ed., 2006.

CSC-253: Design and Analysis of Algorithms **L | T | P (4 | 0 | 2)**

Introduction: Review of Asymptotic Notations, Mathematical analysis for Recursive and Nonrecursive algorithms, solving recurrence relations.

Algorithm Design Techniques: Brute Force– Closest-Pair and Convex-Hull Problems, Exhaustive Search, Traveling Salesman Problem, Assignment problem. Divide and conquer methodology– Merge sort, Quick sort, Binary search, Multiplication of Large Integers, Strassen's Matrix Multiplication, Closest-Pair and Convex-Hull Problems. Greedy strategy– General Approach and problems like Optimal Merge Patterns, Minimum Spanning Trees, Knapsack Problem, Huffman Code, Job sequencing with deadlines, single source shortest path. Dynamic Programming– General Approach, Memoization, Multistage Graph, Matrix-Chain Multiplication, Longest Common Subsequence, Knapsack Problem, Reliability Design, Floyd Warshall algorithm, Computing a Binomial Coefficient, Optimal Binary Search Trees. Iterative improvement – The Simplex Method, The Maximum-Flow Problem – Maximum Matching in Bipartite Graphs, The Stable Marriage Problem.

Limitations of Algorithm Power: Lower-Bound Arguments, Decision Trees, P, NP and NP-Complete Problems, NP-Hard and NP-Complete Problems, Intractability, Cook's Theorem, NP Completeness Reductions. Coping with the Limitations – Backtracking concept and problems like N-Queens problem, Hamiltonian Circuit Problem, Subset Sum Problem; Branch & Bound method and problems like Assignment problem, Knapsack Problem, Traveling Salesman Problem, Approximation Algorithms for NP-Hard Problems.

Textbooks:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", Prentice Hall of India, 3rd ed., 2010.
2. R. C. T. Lee, S. S. Tseng, R. C. Chang, Y. T. Tsai, "Introduction to the Design and Analysis of Algorithms: A Strategic Approach" McGraw Hill, 2006.
3. Anany Levitin, Introduction to the Design and Analysis of Algorithms, Pearson Education, 2007.
4. Sara Baase, Computer Algorithms: Introduction to Design and Analysis, 3rd ed., Addison-Wesley, 2000.
5. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms", University Press, 2nd ed., 2008.
6. Kenneth A. Berman, Jerome Paul, "Algorithms: Sequential, Parallel and Distributed", Cengage Learning, 2004.
7. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, "The Design and Analysis of Computer Algorithms" Pearson Education, 2008.
8. Michael T. Goodrich, Roberto Tamassia, *Algorithm Design*, Wiley, 2002.
9. S. Dasgupta, C. Papadimitriou, and U. Vazirani. *Algorithms*. McGraw-Hill Higher Education, 2006.

CSC-254: Project-2: Web Application Development **(2 credits)**

In this project, students are expected to work in team and apply all their learning from the courses like web technologies, databases etc. (covered in III and IV semesters) into developing a Web Application.

CSS-250: Server-side Web Technologies **L | T | P (0 | 0 | 2)**

CGI-Perl: Introduction to CGI, Perl data structures, control structures, pattern matching and regular expressions, I/P and O/P in Perl, report formatting in perl, perl built in functions, custom functions, references and anonymous data structures, object oriented programming in perl, advanced data manipulation, database programming with perl, perl-CGI programming, web programming with perl script.

PHP and MySQL: Introduction and basic syntax of PHP, decision and looping with examples, PHP and HTML, Arrays, Functions, Browser control and detection, string, Form processing, Files, Advance Features: Cookies and Sessions, Object Oriented Programming with PHP, Basic commands with PHP examples, Connection to server, creating database, selecting a database, listing database, listing table names, creating a table, inserting data, altering tables, queries, deleting database, deleting data and tables, PHP myadmin and database bugs.

Advanced Scripting Languages: JSON– An alternative to XML technology, overview and concepts of JSON, how to implement JSON in web sites, Python– Python Fundamentals.

Servlets: Retrieving information, sending HTML informations, sending multimedia content, session tracking, security, database connectivity, Applet servlet communication, Inter Servlet communication.

ASP: Basics, variables, ASP control, structures, object properties, methods and events, request and response objects, Application, session, cookies and error handling objects. Scripting objects, ASP components, Data store Access, using Record sets and building script components for ASP.

Textbooks:

1. Jeffrey C Jackson, Web Technology– A Computer Science perspective, Pearson Education, 2007.
2. Michael Mcmillan, Perl from the ground up, Tata Mcgraw- Hill, 1999.
3. Chris Ullman, Beginning ASP 3.0, Wrox Press Ltd, 2001.
4. Paul J. Deitel, Harvey Deitel, Abbey Deitel, Internet and World Wide Web How to Program, 5th ed., 2011.
5. Ralph Moseley and M. T. Savaliya, “Developing Web Applications”, Wiley-India.
6. Web Technologies, Black Book, Dreamtech Press.
7. Harwani, “Developing Web Applications in PHP and AJAX”, McGraw Hill.

CSG-251: Programming in Java **L | T | P (4 | 0 | 2)**

Introduction to Java: Java Architecture and Features, Understanding the semantic and syntax, Differences between C++ and Java, Compiling and Executing a Java Program, Variables, Constants, Keywords, Data Types, Operators, Expressions, Comments, Defining & Using Classes, Controlling Access to Class Members, Basic Program, Decision Making Constructs.

Java Methods, Arrays, Strings and I/O: Java Methods (Defining, Scope, Passing Arguments, Type Conversion and Type Checking, Built-in Java Class Methods), Class Constructors, Method Overloading, Class Variables & Methods, final classes, Object class, Garbage Collection, Creating & Using Arrays (One Dimension and Multi-dimensional), Referencing Arrays Dynamically, Java Strings: The Java String class, Manipulating Strings, String Immutability & Equality, Passing Strings To & From Methods, String Buffer Classes. Simple I/O using System.out and the Scanner class, Byte and Character streams, Reading/Writing from console and files.

Inheritance, Interfaces, Packages: Inheritance: (Single Level and Multilevel, Method Overriding, Dynamic Method Dispatch, Abstract Classes), Interfaces and Packages, extending interfaces and packages, Package and Class Visibility, Using Standard Java Packages (util, lang, io, net), Wrapper Classes.

Exception Handling, Threading: Exception types, uncaught exceptions, throw, built-in exceptions, creating your own exceptions; Multi-threading: The Thread class and Runnable interface, creating single and multiple threads, Thread prioritization.

Applets and Event Handling: Java Applets: Introduction to Applets, Writing Java Applets, working with Graphics, Incorporating Images & Sounds. Event Handling Mechanisms, Listener Interfaces, Adapter and Inner Classes.

Textbooks:

1. Herbert Schildt, "Java, The Complete Reference", TMH, 7th ed., 2007.
2. Ken Arnold, James Gosling, David Homes, "The Java Programming Language", 4th ed., 2005.
3. Cay S. Horstmann, Gary Cornell, "Core Java 2 Volume 1 and 2", Prentice Hall, 9th ed., 2012.
4. Bruce Eckel, "Thinking in Java", PHI, 3rd ed., 2002.
5. Paul Deitel, Harvey Deitel, "Java: How to Program", Prentice Hall, 10th ed., 2011.
6. Bert Bates and Kathy Sierra "Head First Java", O'Reilly Media Inc., 2nd ed., 2005.

<u>CSG-252: Database Systems</u> **L | T | P (4 | 0 | 2)**

Introduction: Overview of databases, Data models, DBMS architecture and data independence, History of Database Systems. Entity-Relationship Modeling: Basic concepts, constraints, keys, Design issues, weak entities.

Relational Data Model and Normalization: Relational model concepts, relational constraints, relational algebra, relational calculus. SQL: basic queries, nested subqueries, aggregate functions, null values, database modification commands. Database Design– Functional dependencies, Normalization, Normal form– 1NF, 2NF, 3NF, BCNF.

Integrity and Security: Domain Constraints, Referential Integrity Constraints, Assertions, Triggers, Security and Authorization– Authorization in SQL, Encryption and Authentication.

Transaction and Concurrency Control: Transaction concept, Transaction state, ACID properties. Concurrency Control– Lock Based Protocols, Timestamp Based Protocols, Validation Based Protocols, Multiple Granularity. Recovery System– Failure Classification, Storage Structure, Recovery and Atomicity, Log based recovery.

Textbooks:

1. Ramez Elmasri, Shamkant B. Navathe, "Fundamentals of Database Systems", Pearson Education, 5th ed., 2008.
2. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, "Database Concepts", McGraw-Hill, 6th ed., 2013.
3. R. Ramakrishanan, J. Gehrke, "Database Management Systems" McGraw-Hill, 3rd ed., 2002.
4. Peter Rob, Carlos Coronel, "Database Systems: Design, Implementation and Management", 7th ed., 2006.

CSG-253: Mathematical Methods **L | T | P (5 | 1 | 0)**

Ordinary Differential Equations: Solution of linear differential equations with constant coefficients. Euler-Cauchy equations, Solution of second order differential equations by changing dependent and independent variables. Method of variation of parameters, Introduction to series solution method.

Partial Differential Equations: Formation of first and second order partial differential equations. Solution of first order partial differential equations: Lagrange`s equation, Four standard forms of non-linear first order equations.

Laplace Transform: Laplace and inverse Laplace transform of some standard functions, Shifting theorems, Laplace transform of derivatives and integrals. Convolution theorem, Initial and final value theorem. Laplace transform of periodic functions, error functions, Heaviside unit step function and Dirac delta function. Applications of Laplace transform.

Z-Transform: Z-transform and inverse Z-transform of elementary functions, Shifting theorems, Convolution theorem, Initial and final value theorem. Application of Z- transform to solve difference equations.

Fourier series: Trigonometric Fourier series and its convergence. Fourier series of even and odd functions. Fourier half-range series. Parseval`s identity. Complex form of Fourier series.

Fourier Transforms: Fourier integrals, Fourier sine and cosine integrals. Fourier transform, Fourier sine and cosine transforms and their elementary properties. Convolution theorem. Application of Fourier transforms to BVP.

Textbooks:

1. Kreyszig, E., "Advanced Engineering Mathematics", Johan Wiley & Sons 2011.
2. Jain, R. K. and Iyenger, S. R. K., "Advanced Engineering Mathematics", Narosa Publishing House 2009.
3. Amarnath, T., "An Elementary Course in Partial Differential Equations", Narosa Publishing House, 2^{nd} ed., 2012.
4. Hildebrand F. B., "Methods of Applied Mathematics", Courier Dover Publications 1992.
5. Rao, K. S., "Introduction to Partial Differential Equations", PHI Learning Pvt. Ltd. 2^{nd} ed., 2010.
6. Sneddon, I. N., " Elements of Partial Differential Equations", McGraw-Hill Book Company 1988.
7. Simmons, G. F. and Krantz, S. G., Differential Equations: Theory, Technique and Practice", Tata McGraw-Hill, 2007.
8. H.K. Dass, “Advanced Engineering Mathematics”, S. Chand & Company, 9^{th} Revised ed., 2001.

SEMESTER V

CSC-301: Automata Theory and Computability **L | T | P (3 | 1 | 0)**

Set of alphabet, strings, operations on strings, Levi's theorem, Terminal and non-terminal symbols, Definition of an automaton, Finite automaton (FA) and its description, Transition system, Acceptance by a finite automaton, Deterministic and non-deterministic automaton (DFA and NFA), Equivalence of DFA and NFA, Mealy and Moore machines, Minimization of states in a finite automaton.

Formal languages and grammar, Language generated by a grammar, Chomsky's hierarchy of languages, Regular grammar and regular expression, Construction of a FA for regular expression, Equivalence of two FAs, Equivalence of two regular expressions, Pumping lemma for regular sets.

Push down automaton (PDA), Transition function for PDA, Acceptance by PDA using final state or empty stack, Language for PDA– context free language, Context free grammar and derivation trees, Leftmost and rightmost derivation, Ambiguity in context free grammar, Pumping lemma for context free language.

Turing machine (TM) model, Instantaneous description, Transition table for a TM, Language for a TM, designing of a TM, Universal TM, Non-deterministic TM, TM and type-0 grammar, Halting problem of a TM, Linear bounded automaton (LBA) model, language of a LBA, Context sensitive languages and grammars.

Unsolvable problems, Reduction techniques, Decidability– Post's correspondence problem (PCP), Rice's theorem, Decidability of membership, Emptiness and equivalence problems of languages, The Turing model of computation, The classes P and NP, NP-Completeness; Satisfiability and Cook's theorem; Polynomial reduction and some NP-complete problems, New paradigms of computing– DNA computing, Membrane computing.

Textbooks:

1. KLP Mishra, N. Chandrasekaran, "Theory of Computer Science (Automata, Languages and Computation)", PHI, 3rd ed.
2. Peter Linz, Jones, Bartlett, "An Introduction to Formal Languages and Automata", 5th ed.
3. John E. Hopcroft, J.D. Ullman, Rajiv Motwani, "Introduction to Automata Theory, Languages and Computation", Pearson Education, 3rd ed.
4. Michael Sipser, "Introduction to the Theory of Computation", Cengage Learning, 3rd ed.

CSC-302: Computer Networks **L | T | P (4 | 0 | 1)**

Introduction to Computer Networks and Data Communication: Network definition; network topologies; network classifications; network protocol; layered network architecture; overview of OSI reference model; overview of TCP/IP protocol suite. Analog and digital signal; data-rate limits; digital to digital line encoding schemes; pulse code modulation; parallel and serial transmission; digital to analog modulation; multiplexing techniques– FDM, TDM; transmission media.

Networks Switching Techniques and Access mechanisms: Circuit switching; packet switching– connectionless datagram switching, connection-oriented virtual circuit switching; dial-up modems; digital subscriber line; cable TV for data transfer.

Data Link Layer: Error detection and error correction techniques; data-link control-framing and flow control; Error recovery protocols– stop and wait ARQ, go-back-n ARQ; CSMA/CD protocols; Ethernet LANS; connecting LAN and back-bone networks– repeaters, hubs, switches, bridges, router and gateways.

Network and Transport Layers: Routing; routing algorithms; Network layer protocol of Internet– IP protocol, Internet control protocols, Transport services, Transport protocols, Internet Transport Protocols– UDP, TCP.

Application Layer: Client–server model, Domain Name System (DNS), Dynamic Host Configuration Protocol (DHCP), Telnet, Network Virtual Terminal (NVT), File Transfer Protocol (FTP), Simple Mail Transfer Protocol (SMTP), User Agent (UA), Mail Transfer Agent (MTA), Multipurpose Internet Mail Extensions (MIME), Post Office Protocol (POP), Simple Network Management Protocol (SNMP), Hypertext Transfer Protocol (HTTP), Uniform Resource Locator (URL), World Wide Web (WWW).

Textbooks:

1. B.A. Forouzan, "Data Communications and Networking", THM, 4th ed., 2007.
2. Andrew S. Tanenbaum, "Computer Networks", PHI, 4th ed., 2003.

References:

1. J.F. Kurose, K.W. Ross, "Computer Networking: A Top-Down Approach", Pearson, 6th ed., 2012.
2. Leon Garcia, Widjaja, "Communication Networks: Fundamental Concepts and Key Architectures", Tata McGraw Hill, 2001.

CSC-303: Numerical and Statistical Computing **L | T | P (4 | 0 | 1)**

Computer Arithmetic and Errors: Floating Point Arithmetic, Machine epsilon, Types of errors: Round off Error, Chopping Error, Truncation Error, Associative and Distributive Law in Floating Point arithmetic, Inherent Error, Error propagation, Numerical Instability, Error in the approximation of a function, Error in series approximation.

Equations Solving Methods: Solution of algebraic and transcendental equation using bisection method, Regula-falsi method, Newton-Raphson method, Solution of simultaneous linear equations using Gauss-elimination method, Jacobi's iterative method, Gauss-Seidel iterative method, LU decomposition methods.

Interpolation: Finite difference and operators, Newton forward, Newton backward, central differences, Stirling's interpolation, divided difference formula.

Differentiation and Integration: Numerical differentiation, formula for derivatives, maxima and minima of a tabulated function, Numerical integration: Newton-cotes formula, Trapezoidal rule, Simpson's rule, Weddle's rule.

Solution of Ordinary Differential Equations using Picard's method, Taylor's series method, Euler's method, modified Euler's method, Runge-Kutta method, predictor-corrector method.

Statistical Computing: Curve fitting, Cubic Spline and Approximation– Method of least squares, fitting of straight lines, polynomials, exponential curves, Frequency Chart– Different frequency chart like Histogram, Frequency curve, Pi-chart. Regression analysis– Linear and Non-linear regression, multiple regressions, Statistical Quality Control methods.

Textbooks:

1. S.S. Sastry, "Introductory Methods of Numerical Analysis", 5th ed., PHI, 2012.
2. M.K. Jain, S.R.K. Iyengar, R.K. Jain, "Numerical Methods for Scientific and Engineering Computation", New Age International, 2003.
3. R.S. Gupta, "Elements of Numerical Analysis", Cambridge University Press, 2015.

CSC-304: Software Engineering **L | T | P (3 | 1 | 0)**

Software Engineering Paradigms: Software Characteristics, Software myths, Software Applications, Software Engineering Definitions, Software Process Models, Waterfall, Prototyping, Spiral (including WIN-WIN Spiral), RAD, Process iteration, Process activities, Software Project Management: Management activities, Project planning, Project scheduling, Risk management and activities.

Software Requirements Engineering: Requirements Engineering Processes, Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management. Software Requirements, Functional and non-functional requirements, User requirements, System requirements, Interface specification, Software Requirement Specification (SRS) document. Specification languages.
Software Metrics and Measures— Process Metrics, Project metrics, Software Project Estimation Models- Empirical, Putnam, COCOMO models.

Software Design Process: Principles of software design, Design Strategies, Levels of software design, Interface Design, Coding, Software Reuse. Software Testing, Software Reliability, Software Safety, Defect testing, Debugging Tools.

Maintenance: Types of Maintenance, Maintenance Cost, Software Configuration Management, Software Reuse, Software Evolution, Reverse Engineering, Introduction to legacy systems, Software Quality Assurance– plans & activities, Software Documentation. Role of documentation in maintenance and types of documentation.

Current trends in Software Engineering: Software Engineering for projects & products. Introduction to Web Engineering and Agile process.

CASE Tools: Computer Aided Software Engineering (CASE), Introduction to CASE tools, Types of CASE tools, Project Management Tools, Analysis tools, Design tools, Programming tools, Prototyping tools, Maintenance tools, Advantages and disadvantages of CASE tools.

Textbooks:

1. K.K. Aggarwal, Yogesh Singh, "Software Engineering", New Age International, 2nd ed., 2005.
2. R.S. Pressman, "Software Engineering – A practitioner's approach", McGraw Hill, 5th ed., 2001.
3. Stephen R. Schach, "Classical & Object Oriented Software Engineering", IRWIN, 1996.
4. James Peter, W. Pedrycz, "Software Engineering: An Engineering Approach", John Wiley & Sons.
5. I. Sommerville, "Software Engineering", Addison Wesley, 2002.

CSE-301: Modeling and Simulation **L | T | P (3 | 0 | 1)**

Introduction to System Modeling: The notion of system, model, simulation, Types of simulations, Illustrative examples, Conceptual and computer models, Verification and validation of models, Simulation experiment, Simulation project life cycle, Description of simulation models, Structure vs. behavior models, Classification of tasks solved within the modeling and simulation process, Detailed example introduction: database server as a typical queuing system. Description of discrete-event systems behavior. Modeling of time. The notion of status, event, activity, process and their interdependencies. Object-oriented model design. Simulation time, control of time advancement, event list. Event driven simulation algorithm. Detailed example: implementation of the database server as a queuing system.

Random Numbers in Simulation: Random variables with discrete and continuous probability distribution. Pseudo-random generators. Multiplicative and additive congruential method. Nonuniform random numbers. Testing of pseudo-random generators. Monte Carlo method. Precision. Queueing systems. Entities: queues, service facilities, storages. Properties of input and output stream. Kendall classification of queueing systems. Entity behavior and statistical data sampling during the simulation run.

Markov Model: Discrete and continuous Markov model. Birth-Death processes. Steady-state queueing systems of types M/M/1, M/M/? M/M/m, M/Er/1, Er/M/1 and their variants, Models M/G/1, G/M/1, G/M/m, G/G/1, G/D/1, M+D/D/1. Closed systems and queueing networks, Simulation languages for discrete-event systems. Case study and comparison: Simscript, GPSS, SOL.

Simulation experiments: Preparation and pre-processing of input data. Statistical data collected during the simulation run. Time dependency of statistics. Histograms. Evaluation and interpretation of results. Model validation and verification. Simulation of digital systems. Abstractions levels of digital system description. Models of signals and functions. Structure vs. behavior. Models of components. Models of delays. Digital systems simulators– methods of implementation. Flow of simulation time. Synchronous and asynchronous algorithm of digital systems simulation. Acceleration of simulation run.

Textbooks:

1. A. M. Law, W.D. Kelton, “Simulation, Modeling and Analysis”, McGraw-Hill, 2nd ed., 1991.
2. Frank L. Severance, “System Modeling and Simulation: An Introduction”, Wiley India, 2009.

CSE-302: Operations Research **L | T | P (3 | 1 | 0)**

Introduction: A quantitative approach to decision making, History and definition of Operations Research, Features of Operations Research Approach, Operations Research Approach to solve a problem, Models and Modelling, Advantages of model building, Methodology, Advantages, Opportunities, Shortcomings, Features and Applications of Operations Research.

Linear programming: Structure of Linear Programming, Advantages Limitations, Application areas of Linear Programming. Mathematical formulations of LP models for product mix problems, graphical and simplex method of solving LP problems, sensitivity analysis, duality.

Transportation and Assignment Problem: Various methods (NWCM, LCM, VAM) of finding initial basic feasible solution and optimal cost. Mathematical model of the Assignment Problem, Hungarian Methods for solving assignment problem, Travelling Salesman Problem.

Network Analysis: Network definition and Network diagram, probability in PERT analysis, project time cost trade off, introduction to resource smoothing and allocation.

Game Theory: Concept of game, two person zero sum game, pure and mixed strategy games, saddle point, dominance method Solution Methods for Games without Saddle point.

Inventory Model: Introduction to inventory control, deterministic inventory model, EOQ model with quantity discount.

Replacement & Maintenance Models: Replacement of items, subject to deterioration of items subject to random failure group vs. individual replacement policies.

Sequencing problem: Johnsons algorithm for n jobs and 2 machines, n jobs and 3 machines, two jobs and m machine problems.

Queuing Theory: Introduction, Concepts relating to queuing systems, basic elements of queuing model, role of Poison & exponential distribution, concepts of birth and death process. The Structure of a Queuing Systems, Performance Measures of a Queuing System, Probability Distribution in Queuing System, Classifications, Solutions of Single Queuing Model Models(M/M/1).

Textbooks:

1. H.A. Taha, "Operations Research: An Introduction", Macmillan, New York, 5th ed., 1992.
2. Govindasami Naadimuthu and Richard E. Johnson, Schaum's Outline of Theory and Problems of Operations Research.
3. Hillier, Frederick S., Gerald J. Lieberman, “Introduction to Operations Research”, McGraw Hill Book Company New York, 6th ed., 1995.
4. Levin, Richard I., David S. Rubin, Goel P. Stinsin and Everett S. Gardener, “Quantitative Approaches to Management”, McGraw Hill Book Company, New York, 8th ed., 1992.

CSE-303: Biometrics **L | T | P (3 | 1 | 0)**

Introduction of Biometric traits and its aim, image processing basics, Geometric Transformations, Linear Interpolation, brightness correction, basic image operations, filtering, enhancement, sharpening, edge detection, smoothening, enhancement, thresholding, localization. Fourier Series, DFT, inverse of DFT.

Biometric system, identification and verification. FAR/FRR, system design issues. Positive/negative identification. Biometric system security, authentication protocols, matching score distribution, ROC curve, DET curve, FAR/FRR curve. Expected overall error, EER, biometric myths and misrepresentations. Selection of suitable biometric. Biometric attributes, Zephyr charts, types of multi biometrics. Verification on Multimodel system, normalization strategy, Fusion methods, Multimodel identification.

Biometric system security and vulnerabilities, circumvention, covert acquisition, quality control, template generation, interoperability, data storage. Recognition systems: Face, Signature, Fingerprint, Ear, Iris etc.

Textbooks:

1. Rafael C. Gonzalez, Richard Eugene Woods, *Digital Image Processing using MATLAB*, 2nd ed., Tata McGraw-Hill Education, 2010.
2. Ruud M. Bolle, Sharath Pankanti, Nalini K. Ratha, Andrew W. Senior, Jonathan H. Connell, *Guide to Biometrics*, Springer, 2009.
3. Richard O. Duda, David G.Stork, Peter E. Hart, *Pattern Classification*, Wiley, 2007.

SEMESTER VI

CSC-351: Artificial Intelligence **L | T | P (4 | 0 | 1)**

Introduction and applications of artificial intelligence, Problem solving: State space search, Production system, Problem characteristics, Problem system characteristics, Search techniques: Generate and test, Hill climbing, Best first search, A* algorithm, Problem reduction.

Knowledge and Reasoning: Knowledge acquisition, Knowledge engineer, Cognitive behavior, Knowledge representation: Level of representation, Knowledge representation schemes, Formal logic, Inference Engine, Semantic net, Frame, Scripts. Adversarial search, Optimal and imperfect decisions, Alpha, Beta pruning, Logical agents: Propositional logic, First order logic – Syntax and semantics – Inference in first order logic. Uncertain Knowledge and Reasoning: Uncertainty – Acting under uncertainty – Basic probability notation – Axioms of probability – Baye's rule – Probabilistic reasoning – Making simple decisions.

Expert systems: Definition, Role of knowledge in expert system, Architecture, Expert System Development Life Cycle: Problem selection, Prototype construction, Formalization, Implementation, Evaluation.

Planning and Learning: Planning: Planning problem – Partial order planning – Planning and acting in non-deterministic domains – Learning: Learning decision trees, Knowledge in learning, Neural networks, Reinforcement learning – Passive and active.

PROLOG Programming: Introduction, variables, using rules, Input and Output predicates, Fail and cut predicates, Recursion, Arithmetic operation, Compound object, Dynamic database, Lists, String, File operations.

Textbooks:

1. Elaine Rich, Kevin Knight, "Artificial Intelligence", Tata McGraw Hill.
2. Dan W. Patterson, "Introduction to Artificial Intelligence and Expert Systems", Prentice Hall of India.
3. Nils J. Nilsson, "Principles of Artificial Intelligence", Narosa Publication house.
4. Stuart Russell, Peter Norvig, "Artificial Intelligence: A Modern Approach", Pearson Education, 2^{nd} ed.
5. Winston, Patrick, Henry, "Artificial Intelligence", Pearson Education.
6. Gopal Krishna, Janakiraman, "Artificial Intelligence".

CSC-352: Computer Graphics **L | T | P (3 | 0 | 1)**

Introduction: Computer Graphics– Overview, Basic elements, Animation and Multimedia Applications, Pictures– Representation, Storage and Display, Visualization and Image Processing, RGB color model, Output/Display Devices– Cathode Ray Tube (CRT), Refreshing Display Devices– Raster scan display device– Pixel, Frame Buffer, Color Display, Random scan display device, Plotters, Printers, Digitizers, Tablets, Light Pen, 3D viewing devices, Active and Passive Graphic Devices, Software for Computer Graphics. Lines– Point Plotting Techniques, Points and Lines, Line drawing algorithms– Digital Differential Analyzer (DDA) algorithm, Bresenham's algorithm, Circle and Ellipse drawing algorithms, Region filling algorithms– Boundary Seed Fill algorithm, Flood Fill algorithm.

Transformations: 2D and 3D Transformations– Translation, Rotation, Scaling and other transformations, Matrix Representation of Points, Homogeneous Coordinate System, 2D and 3D Viewing Transformations, Parallel and Perspective Projections, Clipping and Windowing, Line Clipping algorithms– Cohen-Sutherland Line Clipping algorithm, Cyrus-Beck Line Clipping algorithm.

Curves and Surfaces: Curve representation, Polygon representation methods, Bezier curves, Bezier surfaces, Spline representations, B-spline methods, Hidden Surface Removal– Z-buffer algorithm, Back face detection, Binary Space Partitioning (BSP) tree method, Scan Line Coherence algorithm, Hidden Line Elimination.

Surface Rendering: Illumination/Lighting and Shading Models, Surface Lighting Effects, Basic Lighting Models– Ambient Lighting, Diffuse Lighting, Specular Reflection Lighting Model (Phong Specular Reflection Model), combined effect of Ambient, Diffuse and/or Specular Reflection. Shading– Gouraud Shading, Phong Shading Model, Creating Shadowed Objects, Drawing Shadows, Rendering Texture.

Graphics Programming: Graphics Programming using OpenGL, Programming 2D Applications, The OpenGL API, Primitives and Attributes, Color, Viewing, Control Functions, Polygons and Recursion, The Three-Dimensional Gasket, Plotting Implicit Functions, Interaction, Input Devices, Clients and Servers, Display Lists.

Textbooks:

1. F. S. Hill, "Computer Graphics Using OpenGL", Pearson Education, 2nd ed., 2007.
2. Donald D. Hearn, M. Pauline Baker, "Computer Graphics with OpenGL", Pearson Education, 3rd ed., 2004.
3. David Rogers, "Procedural Elements of Computer Graphics", McGraw Hill, 2nd ed., 2001.

CSC-353: Optimization Techniques **L | T | P (4 | 0 | 1)**

Introduction: Optimization, Types of Problems and Algorithms, Review of Linear Algebra and Analysis, Convex Sets and Convex Functions.

Unconstrained Optimization: Basic properties of solutions and algorithms, Global convergence, Basic Descent Methods: Line Search Methods, Steepest Descent and Newton Methods, Modified Newton methods, Globally convergent Newton Method. Nonlinear Least Squares Problem and Algorithms, Conjugate Gradient Methods, Trust-Region Methods.

Constrained Optimization: First Order Necessary Conditions, Second Order Necessary Conditions, Duality, Constraint Qualification. Convex Programming Problem and Duality Linear Programming: The Simplex Method, Duality and Interior Point Methods, Karmarkar's algorithm, Transportation and Network flow problem. Quadratic Programming: Active set methods, Gradient Projection methods and sequential quadratic programming Dual Methods: Augmented Lagrangians and cutting-plane methods Penalty and Barrier Methods Interior Point Methods.

Textbooks:

1. David Luenberger and Yinyu Ye, "Linear and Nonlinear Programming", 3rd ed., Springer, 2008.
2. Fletcher R., "Practical Methods of Optimization", John Wiley, 2000.
3. S.S. Rao, "Engineering Optimization: Theory and Practice", New Age International Pvt. Ltd., New Delhi, 2000.
4. G. Hadley, "Linear Programming", Narosa Publishing House, New Delhi, 1990.
5. K. Deb, "Optimization for Engineering Design: Algorithms and Examples", PHI, 1995.

CSC-354: Project Work/Dissertation **(6 credits)**

This course requires individual/group effort that is overseen by your project supervisor. Weekly meetings will be held to discuss progress and review necessary documents in support of the project. There will be monthly presentations regarding progress of the project. A final presentation followed by viva-voce by external examiner will be held at the end of the semester where the student must submit a project report.

CSE-351: Natural Language Processing **L | T | P (3 | 0 | 1)**

Natural Language Understanding: The study of Language, Applications of NLP, Evaluating Language Understanding Systems, Different levels of Language Analysis, Representations and Understanding, Organization of Natural Language Understanding Systems, Linguistic Background: An outline of English syntax.

Semantics and Knowledge Representation: Semantics and logical form, Linking syntax and semantics, Ambiguity resolution, Strategies for semantic interpretation, Applications like machine translation.

Grammars for Natural Language: Grammars and sentence Structure, Transition Network Grammars, Feature Systems and Augmented Grammars, Auxiliary Verbs and Verb Phrases, Movement Phenomenon in Language, Handling questions in Context-Free Grammars. Human preferences in Parsing, Encoding uncertainty, Deterministic Parser.

Ambiguity Resolution: Statistical Methods, Probabilistic Language Processing, Estimating Probabilities, Part-of Speech tagging, Obtaining Lexical Probabilities, Probabilistic Context-Free Grammars, Best First Parsing. Semantics and Logical Form, Word senses and Ambiguity, Encoding Ambiguity in Logical Form.

Machine Translation: Problems in Machine Translation, Characteristics of Indian Languages, Machine Translation Approaches: Direct Machine Translation, Rule-based Machine Translation: Transfer-based and Interlingua based Machine Translation, Corpus-based Machine Translation: Statistical and Example-based Machine Translation, Semantic or Knowledge-based MT systems.

Textbooks:

1. James Allen, “Natural Language Understanding”, Pearson Education, 2nd ed., 2003
2. L.M. Ivansca, S.C. Shapiro, “Natural Language Processing and Language Representation”.
3. T. Winograd, “Language as a Cognitive Process”, Addison-Wesley.

<u>CSE-352: Digital Marketing</u> **L | T | P (3 | 0 | 1)**

An Introduction to Digital Marketing, Digital Marketing basics (Online Payments, Disability Web Access, Surveys & Forms, Affiliate & Voucher Marketing, Crowdsourcing).

Search Engine Optimization: Understanding Search Engine, Search Engine Result Pages, Search Behavior, On-Page Optimization, Off-Page Optimization,

Digital Display Advertising, Online Advertising, Social Media Marketing (Facebook & LinkedIn), Mastering Google (AdWords Advertising, Analytics & Applications).

Micro Blogging, Twitter, Copy Writing for The Web, Social Media & Mobiles, Mobile Marketing, Email Marketing, Video & Audio (Podcasting) Marketing, Strategic & Action Planning.

Textbooks:

1. Ian Dodson, "The Art of Digital Marketing", Wiley, 2016.

References:

1. A Beginner's Textbook for Digital Marketing online book.

CSE-353: Introduction to Data Science **L | T | P (3 | 0 | 1)**

Introduction: What is Data Science? Big Data and Data Science, Statistical Inference– Populations and samples, Statistical modeling, probability distributions, fitting a model, Intro to R. Exploratory Data Analysis– Basic tools (plots, graphs and summary statistics) of EDA, Philosophy of EDA, The Data Science Process– Case Study: RealDirect (online real estate firm)

Basic Machine Learning Algorithms– Linear Regression, k-Nearest Neighbors (k-NN), k-means, Naive Bayes and why it works for Filtering Spam, Data Wrangling: APIs and other tools for scrapping the Web.

Feature Generation and Feature Selection (Extracting Meaning from Data) – Motivating application: user (customer) retention, Feature Generation (brainstorming, role of domain expertise, and place for imagination) - Feature Selection algorithms – Filters; Wrappers; Decision Trees; Random Forests.

Recommendation Systems: Building a User-Facing Data Product, Algorithmic ingredients of a Recommendation Engine, Dimensionality Reduction, Singular Value Decomposition, Principal Component Analysis, Mining Social-Network Graphs, Social networks as graphs, Clustering of graphs, Direct discovery of communities in graphs, Partitioning of graphs, Neighborhood properties in graphs.

Data Visualization: Basic principles, ideas and tools for data visualization. Data Science and Ethical Issues.

Textbooks:

1. Cathy O'Neil, Rachel Schutt, Doing Data Science: Straight Talk from the Frontline, O'Reilly, 2014.
2. Christopher Bishop, Pattern Recognition and Machine Learning, Springer, 2005.
3. David Barber, Bayesian Reasoning and Machine Learning, Cambridge University Press, 2012.

SEMESTER VII

CSC-401: Probability and Statistical Inference **L | T | P (4 | 0 | 2)**

Introduction to Probability and Statistics: Probability and Randomness, Random Experiment, Sample Space, Random Events, Probability Definition– Axiomatic definition, Frequency Definition. Probability of Non-disjoint events (Theorems). Counting techniques applied to probability problems, Conditional probability, General Multiplication Theorem, Independent events, Bayes' theorem and related problems. Discrete and continuous random variables, Probability mass function, Probability density function. Distributions– Uniform, Exponential, Binomial, Poisson, Normal, t and $\chi 2$. Mean, median, mode, standard deviation, expectation, variance, covariance and correlation. Moment generating function. Binomial approximation to Poisson distribution and Normal distribution. Population, Sample, Central Limit Theorem, Law of large numbers (Weak law), Estimation of parameters, Sampling distribution of sample mean and sample variance, Maximum likelihood estimate of statistical parameters.

Statistical Learning: Regression Models, Dimensionality and Structured Models, Model Selection and Bias-Variance Trade-off, Classification, Introduction to R. Linear Regression, Hypothesis Testing and Confidence Intervals, Multiple Linear regression.

Classification: Introduction to Classification Problems, Logistic Regression, Multivariate Logistic Regression, Discriminant Analysis, Gaussian Discriminant Analysis, Quadratic Discriminant Analysis, Naïve Bayes, Classification in R.

Resampling Methods: Cross-validation, k-fold cross validation, The Bootstrap, Resampling in R. Linear Model Selection and Regularization, Dimension Reduction Methods, Principal Components Regression and Partial Least Squares, Model Selection in R.

Nonlinear Functions: Polynomial and step functions, Piecewise Polynomials and Splines, Local regression, Nonlinear functions in R. Tree Based Methods.

Support Vector Machines: Optimal Separating Hyperplanes, Support Vector Classifier, Feature Expansion and the SVM, SVMs in R.

Unsupervised Learning: Principal Components, Higher Order Principal Components, k-means Clustering, Hierarchical Clustering.

Textbooks:

1. Sheldon Ross, "A First Course in Probability", 9th ed., Pearson Education, 2013.
2. Vijay K. Rohatgi, A. K. Md. Ehsanes Saleh, "An Introduction to Probability and Statistics", 2nd ed., Wiley, 2008.

References:

1. Trevor Hastie, Robert Tibshirani, Jerome Friedman, "The Elements of Statistical Learning: Data Mining, Inference, and Prediction", Springer, 2nd ed., 2017.
2. Vijay K. Rohatgi, "Statistical Inference", 2003.
3. Bradley Efron, Trevor Hastie, "Computer Age Statistical Inference: Algorithms, Evidence and data Science", Cambridge University Press, 2016.

<u>CSC-402: Compiler Design</u> **L | T | P (4 | 0 | 2)**

Compiler Structure: Analysis-synthesis model of compilation, various phases of a compiler, tool based approach to compiler construction. Lexical analysis: Interface with input parser and symbol table, token, lexeme and patterns, difficulties in lexical analysis, error reporting and implementation. Regular grammar & language definition, Transition diagrams, Design of a typical scanner using LEX or Flex.

Syntax Analysis: Context free grammars, ambiguity, associability, precedence, top down parsing, recursive descent parsing, transformation on the grammars, predictive parsing LL(1) grammar, Bottom up parsing, operator precedence grammars, LR parsers (SLR, LALR, LR), Design of a typical parser using YACC or Bison.

Syntax directed definitions: Inherited and synthesized attributes, dependency graph, evaluation order, bottom up and top down evaluation of attributes, L- and S-attributed definitions. Type checking: type system, type expressions, structural and name equivalence of types, type conversion, overloaded function and operators, polymorphic function. Run time system: storage organization, activation tree, activation record, parameter passing symbol table, dynamic storage allocation. Intermediate code generation: intermediate representation, translation of declarations, assignments, Intermediate Code generation for control flow, Boolean expressions and procedure calls, implementation issues.

Code generation and instruction selection: Issues, basic blocks and flow graphs, register allocation, code generation, DAG representation of programs, code generation from DAGS, peep hole optimization, code generator, specification of machine.

Code optimization: Source of optimizations, optimization of basic blocks, loops, global dataflow analysis, solution to iterative dataflow equations, code improving transformations, dealing with aliases, data flow analysis of structured flow graphs.

Textbooks:

1. K. C. Louden, "Compiler Construction, Principle and Practice", Cengage Publication, 6th ed., 2009.
2. Alfred V. Aho, Ravi Sethi, Jeffrey, D. Ullman, "Compilers: Priciples, Techniques and Tools", Pearson, 1998.
3. V. Raghvan, "Principles of Compiler Design", TMH, 2009.
4. Levine, Mason, Brown, "Lex & Yacc", O' Reilly, 1998.
5. S. S. Muchnick Harcourt Asra, "Advanced Compiler Design implementation", Morgan Kaufman, 2006.
6. Allen, "Modern Compiler Implementation in C", Cambridge University Press, 1997.
7. Vinu V. Das, "Compiler Design using FLEX and YACC", PHI, 2005.
8. Cooper, "Engineering a Compiler", Elsevier, 2005.
9. Alan I. Holub, "Compiler Design in C", PHI, 2009.
10. Fisher, "Crafting a Compiler in C", Pearson, 2005.

CSE-401: Computational Number Theory **L | T | P (3 | 0 | 1)**

Elementary Number Theory: Divisibility, The Division Algorithm, Number Patterns, Prime and Composite Numbers, Fibonacci and Lucas Numbers, Fermat Numbers, Greatest Common Divisors, The Euclidean Algorithm, The Fundamental Theorem of Arithmetic, Least Common Multiple, Linear Diophantine Equations

Congruence: Linear Congruence, Pollard's Rho Factoring Method, Congruence Applications, Divisibility Tests, Modular Designs, Check Digits, Systems of Linear Congruence, The Chinese Remainder Theorem, General Linear Systems, Linear Systems.

Primality Testing and Discrete Logarithms: Fermat test, Miller-Rabin test, Solovay-Strassen test, AKS test, Trial division, Pollard rho method, p-1 method, CFRAC method, quadratic sieve method, elliptic curve method, Computing discrete logarithms, Baby-step-giant-step method, Pollard rho method, Pohlig-Hellman method, index calculus methods, linear sieve method, Coppersmith's algorithm.

Finite fields: Prime and extension fields, representation of extension fields, polynomial basis, primitive elements, normal basis, optimal normal basis, irreducible polynomials. Algorithms for polynomials: Root-finding and factorization, Lenstra-Lenstra-Lovasz algorithm, polynomials over finite fields.

Elliptic curves: The elliptic curve group, elliptic curves over finite fields, Schoof's point counting algorithm, Applications: Algebraic coding theory, cryptography.

Textbooks:

1. Thomas Koshy, "Elementary Number Theory with Applications", Academic Press, 2007.
2. David M. Burton, Elementary Number Theory, 7^{th} ed., McGraw Hills, 2012.
3. Ivan Niven Herbert S. Zuckerman Hugh L. Montgomery, An Introduction to the Theory of Numbers, 5^{th} ed., Wiley India Pvt. Ltd., 2015.
4. Abhijit Das, Computational Number Theory, CRC/CH, 2013.
5. Victor Shoup, A Computational Introduction to Number Theory and Algebra, Cambridge University Press, 2008.
6. Alfred J. Menezes, Paul C. van Oorschot, Scott A. Vanstone, Handbook of Applied Cryptography, CRC Press, 1996.

CSE-402: Digital Signal Processing **L | T | P (3 | 0 | 1)**

Introduction: Signals, systems and signal processing, classification of signals, elements of digital signal processing system, concept of frequency in continuous and discrete time signals, Periodic Sampling, Frequency domain representation of sampling, Reconstructions of band limited signals from its samples, general applications of DSP.

Discrete-Time Signals and Systems: Discrete-Time Signals, Discrete-Time Systems, LTI Systems, Properties of LTI Systems, linear convolution and its properties, Linear Constant Coefficient Difference equations. Frequency domain representation of Discrete-Time Signals & Systems, Representation of sequences by Discrete Time Fourier Transform (DTFT), Properties of DTFT and correlation of signals, Fourier Transform Theorems.

The Z-transform and Analysis of Linear Time Invariant System: Z-Transform, Properties of ROC for Z-transform, Inverse Z-transform methods, Z-transform properties, Analysis of LTI systems in time domain and stability considerations. Frequency response of LTI system, Relationship between magnitude & phase, all pass systems, inverse systems, Minimum/Maximum phase systems, systems with linear phase. Structures for Discrete Time Systems: Block Diagram and signal flow diagram representations of Linear Constant-Coefficient Difference equations, Basic Structures of IIR Systems, Transposed forms, Direct and cascade form Structures for FIR Systems, Effects of Co-efficient quantization.

Filter Design Techniques: Design of Discrete-Time IIR filters from Continuous-Time filters- Approximation by derivatives, Impulse invariance and Bilinear Transformation methods; Design of FIR filters by windowing techniques, Illustrative design examples of IIR and filters. Discrete-Fourier Transform: Representation of Periodic sequences: The discrete Fourier series and its Properties Fourier Transform of Periodic Signals, Sampling the Fourier Transform, The Discrete-Fourier Transform, Properties of DFT, Linear Convolution using DFT.

Fast Fourier Transform: FFT-Efficient Computation of DFT, Goertzel Algorithm, radix2 and radix 4 Decimation-in-Time and Decimation-in-Frequency FFT Algorithms. Architecture of DSP Processors– Harward architecture, pipelining, Multiplier-accumulator (MAC) hardware, architectures of fixed and floating point (TMSC6000) DSP processors.

Textbooks:

1. Alan V. Oppenheim, Ronald W. Schafer, "Digital Signal Processing", Pearson, 1st ed., 2015.
2. Sanjit K. Mitra, "Digital Signal Processing: A Computer-based Approach", McGraw-Hill, 4th ed., 2013.
3. Andreas Antoniou, "Digital Filters: Analysis, Design, and Applications", TMH, 2nd ed., 2001.

CSE-403: Data Mining and Warehousing **L | T | P (3 | 0 | 1)**

Evolution of database technology: Introduction to data warehousing and data mining, difference between operational databases and data warehouses.

Data warehouse architecture & design: Data warehousing Components, building a Data warehouse, Mapping the Data warehouse to multiprocessor architecture, DBMS Schema as for Decision Support, Data Extraction, Clean up and Transformation tools, Metadata.

Data mining: Data Pre-processing & Data Mining Primitives Data Pre-processing, Data cleaning, Data Integration and Transformation, Data reduction, Discretization and Concept Hierarchy Generation, Data Mining primitives, Languages and System Architectures, Concept Description: characterization and Comparison, Analytical Characterization, Mining Class Comparison.

Association Rules & Mining Association Rule Mining: Mining of Single dimensional Boolean association rules, Constraint based association Mining Classification and prediction: Basic issues regarding classification and prediction, Classification by Decision Tree, Bayesian classification, Prediction, Classifier accuracy.

Cluster Analysis: Basic issues, clustering using partitioning methods, Hierarchical methods, Density based methods, Grid based methods and model based methods, Algorithms for outlier analysis.

Textbooks:

1. Ralph Kimball, "The Data Warehouse Life Cycle Toolkit", John Wiley & Sons Inc., 1998.
2. Alex Berson, S.J. Smith, "Data Warehousing, Data Mining & OLAP", TMH, 1997.
3. W.H. Inmon, "Building the Data Warehouse", Wiley India, 2011.

CSE-404: Fuzzy Logic **L | T | P (3 | 0 | 1)**

Introduction to Soft Computing: Evolution of Computing, Soft Computing Constituents–From Conventional AI to Computational Intelligence.

Fuzzy Sets and Uncertainty: Uncertainty and information, fuzzy sets and membership functions, chance verses fuzziness, properties of fuzzy sets, fuzzy set operations.

Fuzzy Relations: Cardinality, operations, properties, fuzzy cartesian product and composition, fuzzy tolerance and equivalence relations, forms of composition operation. Fuzzification and Defuzzification: Various forms of membership functions, fuzzification, defuzzification to crisp sets and scalars.

Fuzzy Logic and Fuzzy Systems: Classic and fuzzy logic, approximate reasoning, Natural language, linguistic hedges, fuzzy rule based systems, graphical technique of inference. Development of membership functions: Membership value assignments: intuition, inference, rank ordering, neural networks, genetic algorithms, inductive reasoning.

Fuzzy Arithmetic and Extension Principle: Functions of fuzzy sets, extension principle, fuzzy mapping, interval analysis, vertex method and DSW algorithm. Fuzzy Optimization: One dimensional fuzzy optimization, fuzzy concept variables and casual relations, fuzzy cognitive maps, agent based models.

Fuzzy Control Systems: Fuzzy control system design problem, fuzzy engineering process control, fuzzy statistical process control, industrial applications.

Textbooks:

1. T. J. Ross, "Fuzzy logic with Engineering Applications", 3rd ed. McGraw-Hill, 2011.
2. H. J. Zimmermann, "Fuzzy set theory and its applications", Springer, 4th ed., 2006.
3. George J. Klir and Bo Yuan, "Fuzzy Sets and Fuzzy Logic-Theory and Applications", Prentice Hall, 1995.
4. Klir, G. and Yuan, B., "Fuzzy Set and Fuzzy Logic: Theory and Applications", Prentice Hall, 2002.
5. T. Terano, K. Asai, and M. Sugeno, "Fuzzy systems theory and its applications", 1 ed. San Diego, CA: Academic press, 1992.
6. Jyh-Shing Roger Jang, Chuen-Tsai Sun, Eiji Mizutani, "Neuro-Fuzzy and Soft Computing", Prentice-Hall of India, 2003.

CSE-405: Wireless and Mobile Computing **L | T | P (3 | 0 | 1)**

Introduction: Challenges in Mobile Computing, coping with uncertainties, Resource poorness, Cellular architecture, Mobility Management, Wireless Medium Access Control: Motivation for a specialized MAC (Hidden and exposed terminals, Near and far terminals), SDMA.

Publishing and Accessing Data in Air: Pull and push based data delivery models, Data dissemination by broadcast, broadcast disks, Directory service in air, Energy efficient indexing scheme for push based data delivery, File System Support for Mobility: Distributed file sharing for mobility support, Coda and other storage manager for mobility support.

Ad hoc Network Routing Protocols: Ad hoc network routing protocols, Destination sequenced distance vector algorithm, Cluster based gateway switch routing, Global state routing, Fish-eye state routing, dynamic source routing, Ad hoc on-demand routing, Location aided routing, Zonal routing algorithm.

Mobile Transaction and Commerce: Models for mobile transaction, Kangaroo and Joey transactions, Team transaction, Recovery model for mobile transactions, Electronic payment and Protocols for mobile commerce.

Textbooks:

1. Jochen H. Schiller, "Mobile Communications", Pearson Education, 2nd ed., 2009.
2. Ivan Stojmenovic "Handbook of Wireless Networks and Mobile Computing", Wiley, 2002.
3. Reza B'far, "Mobile Computing Principles: Designing and Developing Mobile Applications with UML and XML", Cambridge University Press, 2004.

CSE-406: Introduction to Cloud Computing **L | T | P (3 | 0 | 1)**

Cloud Computing Fundamentals: Cloud Computing definition, private, public and hybrid cloud. Cloud types; IaaS, PaaS, SaaS. Benefits and challenges of cloud computing, public vs private clouds, role of virtualization in enabling the cloud; Business Agility: Benefits and challenges to Cloud architecture. Application availability, performance, security and disaster recovery; next generation Cloud Applications.

Cloud Applications: Technologies and the processes required when deploying web services; Deploying a web service from inside and outside a cloud architecture, advantages and disadvantages

Management of Cloud Services: Reliability, availability and security of services deployed from the cloud. Performance and scalability of services, tools and technologies used to manage cloud services deployment; Cloud Economics: Cloud Computing infrastructures available for implementing cloud based services. Economics of choosing a Cloud platform for an organization, based on application requirements, economic constraints and business needs (e.g Amazon, Microsoft and Google, Salesforce.com, Ubuntu and Redhat)

Application Development: Service creation environments to develop cloud based applications. Development environments for service development; Amazon, Azure, Google App.

Cloud IT Model: Analysis of Case Studies when deciding to adopt cloud computing architecture. How to decide if the cloud is right for your requirements. Cloud based service, applications and development platform deployment so as to improve the total cost of ownership (TCO).

Textbooks:

1. Gautam Shroff, Enterprise Cloud Computing Technology Architecture Applications.
2. Toby Velte, Anthony Velte, Robert Elsenpeter, Cloud Computing, A Practical Approach.
3. Dimitris N. Chorafas, Cloud Computing Strategies.

SEMESTER VIII

CSC-451: Advanced Algorithms **L | T | P (4 | 0 | 2)**

Review of Analysis Techniques: Growth of Functions: Asymptotic notations; Standard notations and common functions; Recurrences and Solution of Recurrence equations: The substitution method, The recurrence–tree method, The master method; Amortized Analysis: Aggregate, Accounting and Potential Methods.

Advanced Data Structures: Red Black Trees, B-Tree, Augmenting Data Structure, Priority Queues, Binomial Heap, Fibonacci Heap, Mergeable Heaps, Data Structure for Disjoint Sets and Union-Find Algorithm.

String Matching Algorithms: Naïve String Matching, Rabin-Karp, String matching with finite automata, Knuth-Morris-Pratt (KMP) Algorithm, Boyer–Moore algorithm.

Number Theoretic Algorithms: Factorization, GCD, Modular Arithmetic, Solving modular linear equations; The Chinese remainder theorem; Powers of an element; RSA cryptosystem; Primality testing; Integer factorization.

Graph Algorithms: Bellman-Ford Algorithm; Single source shortest paths in a DAG; Johnson's Algorithm for sparse graphs; Flow networks and Ford-Fulkerson method; Maximum bipartite matching.

Probabilistic and Randomized Algorithms: Probabilistic algorithms; Randomizing deterministic algorithms– Randomized Quicksort, Computational Geometry– Convex Hull. Monte Carlo and Las Vegas algorithms; Probabilistic numeric algorithms Approximation Algorithms and Randomized Algorithms– Polynomial Time Approximation Schemes.

Textbooks:

1. T. H Cormen, C E Leiserson, R L Rivest and C Stein: Introduction to Algorithms, 3rd ed., Prentice-Hall of India, 2010.
2. Kenneth A. Berman, Jerome L. Paul: Algorithms, Cengage Learning, 2002.
3. Ellis Horowitz, Sartaj Sahni, S.Rajasekharan: Fundamentals of Computer Algorithms, 2nd ed., Universities press, 2007.

CSC-452: Information Theory and Coding **L | T | P (5 | 1 | 0)**

Information Theory: Uncertainty, Information, Entropy: characterization and related properties, Huffman codes, Robustness of coding techniques, Discrete Memoryless Channel, Mutual Information, Channel Capacity, Shannon's Theorems, Fundamental theorem of information theory, Gaussian Channel, Limits to Communication.

Linear Block Codes: Groups, Fields and Vector Spaces, Construction of Galois Fields of Prime Order, Syndrome Error Detection, Standard Array and Syndrome Decoding, Hamming Codes.

Cyclic Codes: Polynomial Representation of Codewords, Generator Polynomial, Systematic Codes, Generator Matrix, Syndrome Calculation and Error Detection, Decoding of Cyclic Codes.

Structure and Properties of Convolutional Codes: Convolutional Encoder Representation, Tree, Trellis, and State Diagrams, Distance Properties of Convolutional Codes, Punctured Convolutional Codes and Rate Compatible Schemes.

Decoding of Convolutional Codes: Maximum Likelihood Detection, The Viterbi Algorithm, Automatic Repeat Request Strategies: Basic Techniques, Hybrid ARQ.

Textbooks:

1. J. A. Thomas and T. M. Cover: Elements of information theory, Wiley, 2006.
2. J. H. van Lint: Introduction to Coding Theory, 3rd ed., Springer, 1998.
3. F. J. MacWilliams and N.J. Sloane: Theory of Error Correcting Codes, Parts I and II, 1977.
4. D. Stinson: Combinatorial Designs: Constructions and Analysis, Springer, 2003
5. P. J. Cameron and J. H. van Lint: Designs, Graphs, Codes and their Links, Cambridge University Press, 2010.
6. C. Fragouli and E. Soljanin: Network Coding Fundamentals, Now Publisher, 2007.

CSE-451: Cryptography and Network Security **L | T | P (3 | 0 | 1)**

Introduction to cryptography: Private key cryptography, Conventional Encryption models, Classical encryption techniques, Substitution cipher, Transposition cipher, Cryptanalysis, Stereography, Stream and block ciphers, Modern block cipher: principles, Shannon's theory of confusion and diffusion, Fiestal structure, DES, Strength of DES, Triple DES, AES, IDEA, Key distribution, Diffie-Hellman algorithm, Public key cryptography, RSA algorithm, Elliptic curve cryptography, Elgamal cryptosystem.

Message Authentication and Hashing: Authentication requirements, Message Digest Algorithms-MD4, MD5, Hash functions, Security of hash functions, Message Authentication Codes (MAC), Secure hash algorithm (SHA). Digital Signatures: Digital Signatures, Authentication protocols, Digital signature standards.

Authentication Applications: Kerberos and X.509, Public Key Infrastructure (PKI), Concept of Digital Certificate, Types of PKI, Electronic mail security-pretty good privacy (PGP), S/MIME. IP Security: Architecture, Authentication header, encapsulating security payloads, combining security associations, key management.

Web and System Security: Secure socket layer (SSL), Transport layer security, Secure electronic transaction (SET). System Security: Intruders, Intrusion Detection System, Password Management, Viruses and related threats, Distributed Denial of Service Attacks, Firewalls, Firewall design principles, Trusted systems.

Textbooks:

1. William Stallings, Cryptography and Network Security: Principals and Practice, Pearson Education, 6th ed., 2013.
2. B. Forouzan, Cryptography and Network Security, TMH, 2nd ed., 2010.
3. Atul Kahate, Cryptography and Network Security, TMH, 7th ed., 2013.
4. Johannes A. Buchmann, Introduction to Cryptography, Springer, 2nd ed., 2009.
5. Alfred J. Menezes, Paul C. van Oorschot, Scott A. Vanstone, "Handbook of Applied Cryptography", CRC Press, 1996.

CSE-452: Digital Image Processing **L | T | P (3 | 0 | 1)**

Introduction: Examples of fields that use digital image processing, fundamental steps in digital image processing, components of image processing system. Digital Image Fundamentals– A simple image formation model, image sampling and quantization, basic relationships between pixels. Image enhancement in the spatial domain– Basic gray-level transformation, histogram processing, enhancement using arithmetic and logic operators, basic spatial filtering, smoothing and sharpening spatial filters, combining the spatial enhancement methods.

Image restoration: A model of the image degradation/restoration process, noise models, restoration in the presence of noise-only spatial filtering, Weiner filtering, constrained least squares filtering, geometric transforms, Introduction to the Fourier transform and the frequency domain, estimating the degradation function, Color Image Processing.

Image Compression: Fundamentals, image compression models, error-free compression, lossy predictive coding, image compression standards. Morphological Image Processing– Preliminaries, dilation, erosion, open and closing, hit or miss transformation, basic morphologic algorithms.

Image Segmentation: Detection of discontinuous, edge linking and boundary detection, thresholding, region-based segmentation. Object Recognition– Patterns and patterns classes, recognition based on decision-theoretic methods, matching, optimum statistical classifiers, neural networks, structural methods– matching shape numbers, string matching.

Textbooks:

1. Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins, "Digital Image Processing using MATLAB", PHI, 2003.
2. Anil K. Jain, "Fundamentals of Digital Image Processing", Prentice Hall, 1989.
3. Digital Image Processing, Rafael C. González, Richard Eugene Woods, Steven L., Pearson, 2010.

CSE-453: Machine Learning **L | T | P (3 | 0 | 1)**

Introduction: Overview of machine learning, related areas, applications, software tools, Parametric regression: linear regression, polynomial regression, locally weighted regression, numerical optimization, gradient descent, kernel methods.

Generative learning: Gaussian parameter estimation, maximum likelihood estimation, MAP estimation, Bayesian estimation, bias and variance of estimators, missing and noisy features, nonparametric density estimation, Gaussian discriminant analysis, naive Bayes.

Discriminative learning: linear discrimination, logistic regression, logit and logistic functions, generalized linear models, softmax regression, Regularization and its utility: The problem of Overfitting, Application of Regularization in Linear and Logistic Regression

Support vector machines: functional and geometric margins, optimum margin classifier, constrained optimization, Lagrange multipliers, primal/dual problems, KKT conditions, dual of the optimum margin classifier, soft margins, kernels, quadratic programming, SMO algorithm.

Graphical and sequential models: Bayesian networks, conditional independence, Markov random fields, inference in graphical models, belief propagation, Markov models, hidden Markov models, decoding states from observations, learning HMM parameters, Factor analysis, independent component analysis, multidimensional scaling, manifold learning.

Deep Learning: Overview, Convolutional Networks, Recurrent Nets, Autoencoders, Recursive Neural Tensor Nets, Platforms for Deep Learning, Deep Reinforcement Learning.

Textbooks:

1. Alex Smola, S.V.N. Vishwanathan, “Introduction to Machine Learning”, Cambridge University Press, 2008.
2. Christopher Bishop, “Pattern Recognition and Machine Learning”, Springer Verlag, 2006.
3. T. Hastie, R. Tibshirani, J. Friedman, “Elements of Statistical Learning”, Springer, 2001.
4. K. Murphy, “Machine Learning: A Probabilistic Perspective”, MIT Press, 2012.
5. Ethem Alpaydin, "Introduction to Machine Learning", The MIT Press, 2^{nd} ed., 2009.
6. Tom M. Mitchell, "Machine Learning", Tata McGraw-Hill Education, 2013.
7. Francois Chollet, “Deep Learning with Python”, Manning Publications Company, 2017.

CSE-454: Neural Networks **L | T | P (3 | 0 | 1)**

Introduction to Neural Networks: Humans and Computers, Organization of the Brain, Biological Neuron, Features, structure and working of Biological Neural Network, Biological and Artificial Neuron Models, Characteristics of ANN, Comparison of BNN and ANN, Models of neuron, Hodgkin-Huxley Neuron Model, Integrate-and-Fire Neuron Model, Spiking Neuron Model, McCulloch-Pitts Model, Perceptron, Adaline model, Basic learning laws, Topology of neural network architecture, Historical Developments, Potential Applications of ANN.

Basic functional units of ANN for pattern recognition tasks: Basic feedforward, Basic feedback and basic competitive learning neural network. Pattern association, pattern classification and pattern mapping tasks. Linear responsibility X-OR problem and solution. Analysis of pattern mapping networks summary of basic gradient search methods. Pattern storage networks, stochastic networks and simulated annealing, Boltzmann machine and Boltzmann learning.

Backpropagation Networks: Architecture of feed forward network, single layer ANN, multilayer perceptron, back propagation learning, input, hidden and output layer computation, backpropagation algorithm, applications, selection of tuning parameters in BPN, Numbers of hidden nodes, learning. Activation & Synaptic Dynamics– Introduction, Activation Dynamics models, Synaptic Dynamics models, Stability and Convergence, Recall in Neural Networks.

Competitive Learning Neural Networks: Components of CL network, pattern clustering and feature mapping network, ART networks, Features of ART models, character recognition using ART network.

Applications of ANN– Pattern classification, Recognition of Olympic games symbols, Recognition of printed Characters, Neocognitron– Recognition of handwritten characters, NETTALK.

Neuro-Fuzzy Modeling: Adaptive Neuro-Fuzzy Inference Systems, Coactive Neuro-Fuzzy Modeling– Classification and Regression Trees, Data Clustering Algorithms, Rulebase Structure Identification, Neuro-Fuzzy Control, Case studies.

Textbooks:

1. L. Fausett, "Fundamentals of Neural Networks: Architectures, Algorithms & Applications", Prentice-Hall, 1994.
2. James A. Freeman and David M. Skapura, "Neural Networks Algorithms, Applications, and Programming Techniques", Pearson, 2003.
3. B. Yegnanarayana, "Artificial Neural Networks", PHI, 2006.
4. Rajasekaran, Pai "Neural networks, Fuzzy Logic and Genetic Algorithms: Synthesis and Applications", PHI, 2011.
5. Stephen I. Gallant, "Neural Network Learning & Expert Systems", MIT Press, 1995.
6. John Hertz, Anders Krogh, Richard G. Palmer, "Introduction to the theory of Neural Computation", Addison-Wesley, 1991.
7. J.-S.R. Jang, C.-T. Sun, E. Mizutani, "Neuro-Fuzzy and Soft Computing", Pearson, 1996.
8. Haykin, S., Neural Networks - A Comprehensive Foundation, 2nd ed., Macmillan, 1999.

CSE-455: Mobile Ad-hoc Networks **L | T | P (3 | 0 | 1)**

Introduction: Ad-hoc Networks– Definition, Characteristics, Features, Application, Characteristics of Wireless Channel, Ad-hoc Mobility Models– Indoor and Outdoor Models. Medium Access Protocols, MAC Protocols, Design Issues, Goals and Classification, Contention Based Protocols– With Reservation, Scheduling Algorithms, Protocols Using Direction Antennas. IEEE Standards– 802.11a, 802.11b, 802.11g, 802.15. HIPERLAN.

Network Protocols: Routing Protocols– Design Issues, Goals and Classification, Proactive vs. Reactive Routing, Uncast Routing Algorithms, Multicast Routing Hierarchical Routing, Quos Aware Routing.

End-End Delivery and Security: Transport Layer– Design Issues, Classification, Ad-hoc Transport Protocols. Security Issues in Ad-hoc Network– Issues and Challenges, Network Security Attacks, Secure Routing Protocols.

Cross Layer Design and Integration of Ad-hoc for 4G: Cross Layer Design– Need for Cross Layer Design, Cross Layer Optimization, Parameter Optimization, Techniques, Cross Layer Cautionary Perspective. Integration of Ad-hoc with Mobile IP Networks. Mesh Networks, Vehicular Area Networks. Ad Hoc On-Demand Distance-Vector Protocol– Properties, Unicast Route Establishment, Multicast Route Establishment, Broadcast, Optimizations and Enhancements.

Link Reversal Routing: Gafni-Bertsekas Algorithm, Lightweight Mobile Routing Algorithm, Temporally Ordered Routing Algorithm, Preserving Battery Life of Mobile Nodes, Associativity Based Routing, Effects of Beaconing on Battery Life, Research Paper on Recent Trends in MANET.

Textbooks:

1. C-K Toh, "Ad Hoc Mobile Wireless Networks: Protocols and Systems", Pearson, 1st ed., 2007.
2. C. Siva Ram Murthy, B. S. Manoj, "Ad Hoc Wireless Networks: Architectures and Protocols", Prentice Hall, 2004.
3. Stefano Basagni, Marco Conti, Silvia Giordano, Ivan Stojmenovic, "Mobile Ad Hoc Networking", Wiley, 2010.
4. Azzedine Boukerche, "Algorithms and Protocols for Wireless, Mobile Ad Hoc Networks", Wiley-Blackwell, 2008.
5. Yi Pan, Yang Xiao, "Ad Hoc and Sensor Networks", Nova Science Publishers, 2005.
6. Carlos de Morais Cordeiro, Dharma Prakash Agrawal, "Ad Hoc and Sensor Networks: Theory and Applications", World Scientific, 2nd ed., 2013.

CSE-456: Cloud Architecture **L | T | P (3 | 1 | 0)**

Distributed System Models and Enabling Technologies: Scalable Computing Service over the Internet. Technologies for Network-based Computing. System Models for Distributed and Cloud Computing. Software Environments for Distributed Systems and Clouds. Performance, Security, and Energy-Efficiency.

Computer Clusters for Scalable Computing: Clustering for Massive Parallelism. Computer Clusters and MPP Architectures. Design Principles of Computer Clusters. Cluster Job and Resource Management. Case Studies of Supercomputers and MPP Systems.

Virtual Machines & Clusters and Data Centre Virtualization: Implementation Levels of Virtualization. Virtualization Structures/Tools and Mechanisms. Virtualization of CPU, Memory and I/O Devices. Virtual Clusters and Resource Management. Virtualization for Data Center Automation.

Design of Cloud Computing Platforms: Cloud Computing and Service Models. Data Centre Design and Interconnection Networks. Architecture Design of Compute and Storage Clouds, Public Cloud Platforms: GAE, AWS and Windows Azure. Cloud Resource Management and Exchanges, Cloud Security and Trust Management.

Challenges in Cloud and Parallel Computing: Review of challenges in Cloud and Parallel Computing with particular emphasis in the areas of Cloud Security, Infrastructure Provisioning, Governance and Compliance, High-Performance Computing, Structured Parallelism, Data-intensive Applications, and Cloud Economics.

Textbooks:

1. Kai Hwang, Jack Dongarra, and Geoffrey C. Fox 2011, Distributed and Cloud Computing: From Parallel Processing to the Internet of Things: Clusters, Grids, Clouds, and the Future Internet, Morgan Kaufmann.
2. Rajkumar Buyya 2013, Mastering Cloud Computing: Foundations and Applications Programming, First Ed., Morgan Kaufmann Waltham, USA.
3. Dan C. Marinescu., Cloud computing, Elsevier/Morgan Kaufmann Boston.
4. San Murugesan (Editor), Irena Bojanova (Editor) 2015, Encyclopedia on Cloud Computing, First Ed., Wiley-Blackwell
5. NIST 2013, Cloud Computing Synopsis and Recommendations, CreateSpace Independent Publishing Platform.

SEMESTER IX

CSC-501: Parallel and Distributed Computing **L | T | P (4 | 0 | 2)**

Introduction: Parallel Computing Architectures, Paradigms, Issues, & Technologies, architectures, topologies, organizations, Parallel Programming Using Shared Memory, memory coherence, race conditions and deadlock detection, synchronization, multithreaded programming, Parallel Programming using Message Passing, synchronous/asynchronous messaging, partitioning and load-balancing.

Advanced Processors and Interconnects: Multicore Processors and High-bandwidth Networks, Parallel and distributed architectures, Distributed and parallel algorithms, Fundamental problems in parallel and distributed computing, fundamental concepts and reasoning principles for parallel and distributed computer systems.

Distributed Programming Algorithms: Fundamental issues and concepts, synchronization, mutual exclusion, termination detection, clocks, event ordering, locking, CORBA, JavaRMI, Web Services, shared spaces.

Clusters of Computers: Server Clusters, High Availability, and Disaster Recovery, synchronization, fault tolerance, coordination and consensus, Virtual Machines and Virtualized Datacenters.

Peer-to-Peer Computing: P2P systems, Familiarity with concurrent programming primitives (semaphores, locks, monitors), Overlay networks, and Content Distribution.

Computational Grids and Applications: National or global computing Grids and Applications.

Textbooks:

1. M J Quinn, Parallel Programming in C with MPI and OpenMP.
2. Ananth Grama, George Karypis, Vipin Kumar, and Anshul Gupta, Introduction to Parallel Computing, 2nd ed., 2003.
3. David Kirk, Wen-Mei W. Hwu, Wen-mei Hwu, Programming Massively Parallel Processors: a hands-on approach, Morgan Kaufmann, 2010.
4. William Gropp, Ewing Lusk, and Anthony Skjellum, Using MPI: Portable Parallel Programming with the Message-Passing Interface, 2nd ed., 1999.
5. Norm Matloff, Programming on Parallel Machines: GPU, Multicore, Clusters and More.
6. K. Hwang and Z. Xu, Scalable Parallel Computing, McGraw-Hill, 1998.
7. G. Coulouris, J. Dollimore, Distributed Systems Concepts and Design, Addison Wesley.
8. Ian Taylor: From P2P to Web Services and Grids, Springer-Verlag, 2005.
9. F. Berman, G. Fox, and T. Hey (Editors), Grid Computing, Wiley, 2003.
10. Hariri and Parashar, Tools and Environments for Parallel & Distributed Computing, John Wiley, 2004.

CSC-502: Combinatorics and Graph Theory **L | T | P (5 | 1 | 0)**

Elementary Concepts in Combinatorics: Basic counting principles, Binomial theorem; Bijective proofs, Combinatorial identities, Permutations of multisets, Multinomial Theorem, Combinations of Multiset, Sterling's Formula, Generalization of Binomial coefficient, Pigeon hole principle and resolution refutation lower bound. Double counting, Matching and Hall's theorem, Inclusion exclusion principle, Inclusion exclusion principle. Solving recurrence relations using generating functions, Partition Number, Catalan Numbers, Sterling numbers of the 2nd kind, Difference Sequences.

Graph Theory: Matchings, Path Cover, Connectivity, Vertex Coloring, Edge Coloring, Other Coloring Problems, Perfect graphs, Planar Graphs, Other special classes of Graphs. Network flow, Introduction to Minor Theory,

The Probabilistic Method: Basics, Markov, Chebishey Inequalities, Lovaz Local Lemma, Linearity of Expectation; The deletion method; The entropy function; Random walks and randomized algorithm for CNF formulas, Random graph.

Spectral Graph theory: Basic properties of graph spectrum; Cheeger's inequality and approximation of graph expansion; Expander graphs and applications to super concentrators and pseudo randomness; Error correcting codes and expander codes; Small set expansion, Unique Games Conjecture and Hardness of approximation.
Additive Combinatorics: Sum product theorem, Szemeredi-Trotter theorem, Kakeya set problem and applications to randomness extractors.

Textbooks:

1. Statys Jukna, Extremal Combinatorics: With Applications in Computer Science, Springer, 2nd ed., 2013.
2. R.P. Grimaldi, B.V. Ramana, Discrete and Combinatorial mathematics – An applied introduction, Pearson Education (2007).
3. Richard A Brnaldi, Introductory Combinatorics, Pearson Education, Inc. (2004).
4. Miklos Bona, Introduction to Enumerative Combinatorics, McGraw Hill (2007).
5. A walk through Combinatorics – An introduction to enumeration and graph theory, World Scientific Publishing Co. Pvt. Ltd. (2006).
6. J.H. Vanlint, R.M. Wilson, A course in Combinatorics, Cambridge University Press.
7. R. Diestel, "Graph Theory", Springer, 2nd ed., 2000.
8. N. Alon and J. Spenser, "Probabilistic Methods", John Wiley and Sons, 2nd ed., 2000.

CSC-503: Project **(4 credits)**

This course requires individual effort that is overseen by your project supervisor. Weekly meetings will be held to discuss progress and review necessary documents in support of the project. There will be monthly presentations regarding progress of the project. A final presentation followed by viva-voce by external examiner will be held at the end of the semester where the student must submit a project report.

CSE-501: Cyber Forensics **L | T | P (3 | 0 | 1)**

History of Forensics: Computer Forensic Flaws and Risks, Rules of Computer Forensics, Legal issues, Digital Forensic Principles, Digital Environments, Digital Forensic Methodologies.

Computer Crime: Introduction to Traditional Computer Crime, Traditional problems associated with Computer Crime. Introduction to Identity Theft & Identity Fraud. Types of CF techniques, Incident and incident response methodology, Forensic duplication and investigation.

Preparation for IR: Creating response tool kit and IR team, Forensics Technology and Systems, Understanding Computer Investigation, Data Acquisition, Evidence Collection: Processing Crime and Incident Scenes, Working with Windows and DOS Systems, Computer Forensics Tools: Software/ Hardware Tools.

Data Forensics: Recovering deleted files and deleted partitions, deleted file recovery tools, deleted partitioned recovery tools, data acquisition and duplication, data acquisition tools, hardware tools, backing up and duplicating data.

E-Mail Forensics and Steganography: Forensics Acquisition, Processing Local mail archives, Processing server level archives, classification of steganography, categories of steganography in Forensics, Types of password cracking.

Validating Forensics Data: Data Hiding Techniques, Performing Remote Acquisition, Network Forensics, Email Investigations, Cell Phone and Mobile Devices Forensics

Textbooks:

1. Kevin Mandia, Chris Prosise, "Incident Response and Computer Forensics", Tata McGraw Hill, 2006.
2. Peter Stephenson, "Investigating Computer Crime: A Handbook for Corporate Investigations", Sept 1999.
3. Anthony Reyes, Jack Wiles, "Cybercrime and Digital Forensics", Syngress Publishers, Elsevier 2007.
4. John Sammons, "The Basics of Digital Forensics", Elsevier 2012
5. Linda Volonins, Reynalds Anzaldua, "Computer Forensics for dummies", Wiley Publishing 2008.
6. Nelson, Phillips, Enfinger, Steuart, "Computer Forensics and Investigations", Cengage Learning, 2008.
7. R.Vacca, "Computer Forensics", Firewall Media, 2005.
8. Richard E. Smith, "Internet Cryptography", Pearson Education, 3rd ed., 2008.
9. Marjie T. Britz, "Computer Forensics and Cyber Crime: An Introduction", Pearson Education, 1st ed., 2012.

CSE-502: Computer Vision and Pattern Recognition **L | T | P (3 | 0 | 1)**

Depth estimation and Multi-camera views: Perspective, Binocular Stereopsis: Camera and Epipolar Geometry; Homography, Rectification, DLT, RANSAC, 3-D reconstruction framework; Auto-calibration. apparel.

Feature Extraction: Edges - Canny, LOG, DOG; Line detectors (Hough Transform), Corners - Harris and Hessian Affine, Orientation Histogram, SIFT, SURF, HOG, GLOH, Scale-Space Analysis- Image Pyramids and Gaussian derivative filters, Gabor Filters and DWT. Image Segmentation: Region Growing, Edge Based approaches to segmentation, Graph-Cut, Mean-Shift, MRFs, Texture Segmentation; Object detection.

Pattern Analysis: Clustering: K-Means, K-Medoids, Mixture of Gaussians, Classification: Discriminant Function, Supervised, Un-supervised, Semi-supervised; Classifiers: Bayes, KNN, ANN models; Dimensionality Reduction: PCA, LDA, ICA; Non-parametric methods.

Motion Analysis: Background Subtraction and Modeling, Optical Flow, KLT, Spatio-Temporal Analysis, Dynamic Stereo; Motion parameter estimation.

Shape from X: Light at Surfaces; Phong Model; Reflectance Map; Albedo estimation; Photometric Stereo; Use of Surface Smoothness Constraint; Shape from Texture, color, motion and edges.

Textbooks:

1. David A. Forsyth, Jean Ponce, Computer Vision: A Modern Approach, 2nd ed., Pearson.
2. R.C. Gonzalez, M.G. Thomason, Syntactic Pattern Recognition: An introduction.
3. P.A. Devijver, J. Kittler, Pattern Recognition: A Statistical Approach.
4. R.O. Duda, P.E. Hart, Pattern Classification and Scene Analysis, Wiley.

CSE-503: Big Data Analytics **L | T | P (3 | 0 | 1)**

Introduction: Big Data and its Importance, Four V's of Big Data, Drivers for Big Data, Introduction to Big Data Analytics, Big Data Analytics applications.

Big Data Technologies: Hadoop's Parallel World, Data discovery, Open source technology for Big Data Analytics, Cloud and Big Data, Predictive Analytics, Mobile Business Intelligence and Big Data, Crowd Sourcing Analytics, Inter- and Trans-Firewall Analytics, Information Management.

Processing Big Data: Integrating Disparate Data Stores, Mapping Data to the Programming Framework, Connecting and Extracting Data from Storage, Transforming Data for Processing, Introduction to MapReduce/Hadoop for analyzing unstructured data, Subdividing Data in Preparation for Hadoop Map Reduce.

Hadoop MapReduce: Employing Hadoop Map Reduce, Creating the components of Hadoop MapReduce jobs, distributing data processing across server farms, Executing Hadoop MapReduce jobs, Monitoring the progress of job flows, The Building Blocks of Hadoop Map Reduce, Distinguishing Hadoop Daemons-Investigating the Hadoop Distributed File System, selecting appropriate execution modes: local, pseudo-distributed, fully distributed.

Advanced Analytics Platform: Real-Time Architecture– Orchestration and Synthesis Using Analytics Engines, Discovery using Data at Rest, Implementation of Big Data Analytics, Big Data Convergence, Analytics Business Maturity Model. HADOOP ECO-SYSTEM: Pig– Installing and Running, Comparison with Databases – Pig Latin – User-Define Functions – Data Processing Operators – Installing and Running Hive– Hive QL – Tables – Querying Data – User-Defined Functions. Fundamentals of HBase and ZooKeeper - IBM InfoSphere BigInsights and Streams. Visualizations– Visual data analysis techniques, interaction techniques; Systems and applications.

Textbooks:

1. Michael Minelli, Michele Chambers, "Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Business", 1st ed., Wiley CIO Series, 2013.
2. Arvind Sathi, "Big Data Analytics: Disruptive Technologies for Changing the Game", 1st ed., IBM Corporation, 2012.
3. Bill Franks, "Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics", 1st ed., Wiley and SAS Business Series, 2012.
4. Noreen Burlingame, Little Book of Big Data, 2012.
5. Tom White, "Hadoop: The Definitive Guide", 3rd ed., O'Reilly, 2012.

<u>CSE-504: Evolutionary Algorithms</u> **L | T | P (3 | 0 | 1)**

Genetic Algorithms: Historical development, GA concepts – encoding, fitness function, population size, selection, crossover and mutation operators, along with the methodologies of applying these operators. Binary GA and their operators, Real Coded GA and their operators

Particle Swarm Optimization: PSO Model, global best, Local best, velocity update equations, position update equations, velocity clamping, inertia weight, constriction coefficients, synchronous and asynchronous updates, Binary PSO.

Differential Evolution: DE as modified GA, generation of population, operators and their implementation.

Genetic programming (GP): Steps in GP, individual representation, initial population, tree creation methods, fitness assessment, individual section methods, GP operators, GP parameters

Introduction to parallel genetic programming, distributed genetic programming, parallel distributed GP.

Textbooks:

1. Gen, M. and Cheng, R. "Genetic Algorithms and Engineering Design", Wiley, New York.
2. David E. Goldberg, "Genetic Algorithm in Search, Optimization and Machine Learning".
3. Wolfgang Banzhaf, Peter Nordin, Robert E. Keller, Frank D. Francone, "Genetic programming: An introduction– On the Automatic Evolution of Computer Programs and its Applications", Morgan Kauffman.

CSE-505: Software Defined Networks **L | T | P (3 | 0 | 1)**

Introduction: History and Evolution of Software Defined Networking (SDN), Separation of Control Plane and Data Plane, IETF Forces, Active Networking. Control and Data Plane Separation: Concepts, Advantages and Disadvantages, the Open Flow protocol.

Network Virtualization: Concepts, Applications, Existing Network Virtualization Framework (VMWare and others), Mininet based examples.

Control Plane: Overview, Existing SDN Controllers including Floodlight and Open Daylight projects. Customization of Control Plane, Switching and Firewall Implementation using SDN Concepts.

Data Plane: Software-based and Hardware-based; Programmable Network Hardware. Programming SDNs: Northbound Application Programming Interface, Current Languages and Tools, Composition of SDNs. Network Functions Virtualization (NFV) and Software Defined Networks: Concepts, Implementation and Applications.

Data Center Networks: Packet, Optical and Wireless Architectures, Network Topologies. Use Cases of SDNs: Data Centers, Internet Exchange Points, Backbone Networks, Home Networks, Traffic Engineering. Use Cases and Looking Forward.

Textbooks:

1. SDN: Software Defined Networks, an Authoritative Review of Network Programmability Technologies, By Thomas D. Nadeau, Ken Gray Publisher: O'Reilly Media, 2013.
2. Software Defined Networks: A Comprehensive Approach, by Paul Goransson and Chuck Black, Morgan Kaufmann, 2014, eBook.
3. Paul Göransson, Chuck Black, Software Defined Networks: A Comprehensive Approach, Elsevier, 2014.
4. Thomas D. Nadeau, SDN: Software Defined Networks, 1st ed., O'reilly.
5. Siamak Azodolmolky, "Software Defined Networking with Open Flow", Packt Publishing, 2013.
6. Fei Hu, "Network Innovation through Open Flow and SDN: Principles and Design", CRC Press, 2014.

CSE-506: Cloud Security **L | T | P (3 | 1 | 0)**

Security Concepts: Confidentiality, privacy, integrity, authentication, non-repudiation, availability, access control, defense in depth, least privilege, how these concepts apply in the cloud, what these concepts mean and their importance in PaaS, IaaS and SaaS. User authentication in the cloud; Cryptographic Systems: Symmetric cryptography, stream ciphers, block ciphers, modes of operation, public-key cryptography, hashing, digital signatures, public-key infrastructures, key management, X.509 certificates, OpenSSL.

Multi-tenancy Issues: Isolation of users/VMs from each other. How the cloud provider can provide this; Virtualization System Security Issues: e.g. ESX and ESXi Security, ESX file system security, storage considerations, backup and recovery; Virtualization System Vulnerabilities: Management console vulnerabilities, management server vulnerabilities, administrative VM vulnerabilities, guest VM vulnerabilities, hypervisor vulnerabilities, hypervisor escape vulnerabilities, configuration issues, malware (botnets etc.)

Virtualization System-Specific Attacks: Guest hopping, attacks on the VM (delete the VM, attack on the control of the VM, code or file injection into the virtualized file structure), VM migration attack, hyper jacking.

Technologies for Virtualization-Based Security Enhancement: IBM security virtual server protection, virtualization-based sandboxing; Storage Security: HIDPS, log management, Data Loss Prevention. Location of the Perimeter.

Legal and Compliance Issues: Responsibility, ownership of data, right to penetration test, local law where data is held, examination of modern Security Standards (e.g. PCIDSS), how standards deal with cloud services and virtualization, compliance for the cloud provider vs. compliance for the customer.

Textbooks:

1. Jared Carstensen, Bernard Golden and JP Morgenthal 2012, Cloud Computing: Assessing the Risks, IT Governance Publishing.
2. J.R. ("Vic") Winkler, Securing the Cloud, Syngress.
3. David G. Rosado, Daniel Mellado, Eduardo Fernandez-Medina and Mario Piattini 2013, Security Engineering for Cloud Computing Approaches and Tools, Information Science Reference.
4. Ronald L. Krutz, Russell Dean Vines, Cloud Security, Wiley.
5. Kunjal Trivedi and Keith Pasley 2012, Cloud Computing Security, Cisco Press.
6. Tim Mather, Subra Kumaraswamy, Shahed Latif, Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, O'Reilly Media, Inc.
7. Charles P. Pfleeger and Shari Lawrence Pfleeger 2007, Security in Computing, Pearson Education, Inc.

SEMESTER X

CSC-551: Research Methodology **L | T | P (4 | 0 | 0)**

Introduction to Research Methods in science – Philosophy of Science, Research methods and Creative Thinking, Evolutionary Epistemology, Scientific Methods, Hypotheses Generation and Evaluation, Code of Research Ethics, Definition and Objectives of Research, Various Steps in Scientific Research, Research presentations Types of Research – Research Purposes – Research Design – Survey Research – formulation of scientific problems and hypotheses – selection of methods for solving a scientific problem Case Study Research.

How to perform a literature review – Sampling Methods – Data Processing and Analysis strategies - Data Analysis with Statistical Packages – Statistical Analysis – Hypothesis-testing – Generalization and Interpretation.

Research Reports - Structure and Components of Research Report, Types of Report, Layout of Research Report, Mechanism of writing a research report – Requirements of a good dissertation.

Textbooks:

1. Oates, B.J., (2005). Researching Information Systems and Computing. Sage Publications, UK.
2. Zobel, J. (2004). Writing for Computer Science - The art of effective communication. 2nd ed., Springer, UK.
3. Garg, B.L., Karadia, R., Agarwal, F. and Agarwal, U.K., 2002. An introduction to Research Methodology, RBSA Publishers.
4. Kothari, C.R., 1990. Research Methodology: Methods and Techniques. New Age International.
5. Sinha, S.C. and Dhiman, A.K., 2002. Research Methodology.

CSC-552: Master's Thesis **(18 Credits)**

The preparation and defense of Master's thesis is the culminating experience in the M.Sc. program. Students are expected to work with a high level of self-motivation. This course requires individual effort that is overseen by your Thesis Advisor. Weekly or bi-weekly meetings will be held to discuss progress and review submitted documents. There will be monthly presentations regarding progress in the thesis work. Once the research and necessary analysis and results compilation are completed, then individual thesis chapters will be written. A final presentation followed by viva-voce by external examiner will be held at the end of the semester where the student is expected to defend his/her thesis. The thesis will be approved upon successful completion of the defense.