Computer Science and Engineering

M. Tech Course

July, 2018

(Last updated: June 2019)

PART I: COURSE STRUCTURE

First Year
Semester I

A. Theory							
Sl.	Course Number	Subject	Scheme Of Studies Per Week			Total	Credits
			L	T	P		
1	CSEN5101	Advanced Data Structures	3	0	0	3	3
2	CSEN5102	Research Methodology and IPR	2	0	0	2	2
3	MATH5101	Advanced Discrete Mathematics and Statistical Methods	3	0	0	3	3
4	CSEN5131 – CSEN5140	Professional Elective I	3	0	0	3	3
	CSEN5131 CSEN5132 CSEN5133 CSEN5134 CSEN5135	Machine Learning Advanced Wireless and Mobile Networks Introduction to Intelligent Systems GPU Computing Image Processing					
5	CSEN5141 – CSEN5150	Professional Elective II	3	0	0	3	3
	CSEN5141 CSEN5142 CSEN5143 CSEN5144 CSEN5145	Data Science Distributed Systems Wireless Sensor Networks Digital Forensics Computational Biology					
6	Audit Course DIMA5116 INCO5117 PDLS5118 YOGA5119 SANS5120	 Disaster Management Constitution of India Personality Development through Life Enlightenment Skills Stress Management by Yoga Sanskrit for Technical Knowledge	2	0	0	2	0
Total Theory			**16**	**0**	**0**	**16**	**14**
Practical							
1	CSEN5151	Advanced Data Structures Lab	0	0	4	4	2
2	CSEN5181 - CSEN5190	Professional Elective-I Lab	0	0	4	4	2
	CSEN5181 CSEN5182 CSEN5183 CSEN5184 CSEN5185	Machine Learning Lab Advanced Wireless and Mobile Networks Lab Introduction to Intelligent Systems Lab GPU Computing Lab Image Processing Lab					
Total Practical			**0**	**0**	**8**	**8**	**4**
Total Semester			**16**	**0**	**8**	**24**	**18**

First Year
Semester II

A. Theory							
Sl.	Course Number	Subject	Scheme Of Studies Per Week			Total	Credits
			L	T	P		
1	CSEN5201	Advanced Algorithms	3	0	0	3	3
2	CSEN5202	Soft Computing	3	0	0	3	3
3	CSEN5231 – CSEN5240	Professional Elective III	3	0	0	3	3
	CSEN5231 CSEN5232 CSEN5233 CSEN5234 CSEN5235	Data Preprocessing and Analysis Secure Software Design and Enterprise Computing Computer Vision Theory of Computation Computational Geometry					
4	CSEN5241 – CSEN5250	Professional Elective IV	3	0	0	3	3
	CSEN5241 CSEN5242 CSEN5243 CSEN5244 CSEN5245	Human and Computer Interaction Graph Algorithms Cloud Computing Algorithms for VLSI CAD Spatial Informatics and GIS					
5	CSEN5231 - CSEN5250	Audit Course – any one subject from Elective III or Elective IV bucket	3	0	0	3	0
Total Theory			**15**	**0**	**0**	**15**	**12**
Practical							
1	CSEN5251	Advanced Algorithms Lab	0	0	4	4	2
2	CSEN5252	Soft Computing Lab	0	0	4	4	2
Total Practical			**0**	**0**	**8**	**8**	**4**
C. Sessional							
1	CSEN5293	Term Paper and Seminar	0	0	4	4	2
Total Semester			**15**	**0**	**12**	**27**	**18**

Second Year
Semester III

A. Theory							
Sl.	Course Number	Subject	Scheme Of Studies Per Week			Total	Credits
			L	T	P		
1	CSEN6131 - CSEN6139	Professional Elective V	3	0	0	3	3
	CSEN6131 CSEN6132 CSEN6133 CSEN6134 CSEN6135 CSEN6136 CSEN6137 CSEN6138 CSEN6139	Mobile Applications and Services Compiler for HPC Computational Complexity Fault Tolerant Computing Approximation Algorithms Randomized Algorithms Information Retrieval Social Network Analysis Quantum Computing					
2		Open Elective	3	0	0	3	3
	CSEN6121 ECEN6122 INFO6123 ECEN6124 MATH6121 AEIE6122	Business Analytics Design of Embedded Systems Information Theory and Coding Automation in VLSI Design Optimization Techniques Intelligent Control					
Total Theory			**6**	**0**	**0**	**6**	**6**
B. Sessional							
1	CSEN6195	Dissertation – Phase I	0	0	20	20	10
Total Semester			**6**	**0**	**20**	**26**	**16**

Second Year
Semester IV

A. Sessional							
Sl.	Course Number	Subject	Scheme Of Studies Per Week			Total	Credits
			L	T	P		
1	CSEN6295	Dissertation – Phase II	0	0	28	28	14
2	CSEN6297	Comprehensive Viva-voce	0	0	0	0	2
Total Semester			**0**	**0**	**28**	**28**	**16**

PART II: DETAILED SYLLABUS

M. Tech. Detailed Syllabus - Semester I

Course Name : Advanced Data Structure					
Course Code: CSEN5101					
Contact Hours Per Week	L	T	P	Total	Credit Points
	3	0	0	3	3

Course Outcomes:

On completion of the course the students undergoing this course are able to:

1. Remember definitions and notations of basic terminologies used in data structures.
2. Learn and understand abstract data types and its significance; differentiate between linear and non-linear data structures for solving real world problems.
3. Understand and apply some of the special trees, Tries data structure and various Hashing Techniques
4. Design modular algorithms on linear and non linear data structures for solving engineering problems efficiently.
5. Understand and analyze the basic principles of different string matching algorithms and identify their advantages and disadvantages.
6. Evaluate the performance of different data structures with respect to various applications.

Module I: Review of Fundamental Concepts (9L)

Introduction: Arrays, lists, stacks, queues, heaps, priority queues, Dictionary operations, Abstract Data Types (ADTs).

Hashing: Hash tables, hash functions, collision resolution by chaining, Collision resolution by open addressing, linear and quadratic probing, Double hashing, extensions and recent advances.

Module II: Binary Trees (9L)

BasicFeatures and Procedures: Binary tree traversal methods, total path length in binary trees, Binary search trees, insertion and deletion of keys, Worst-case and average case times for search, insertion and deletion.

Module III: Other Data Structures for Storage and Search (9L)

B-Trees: Broad shallow tree structures for secondary storage, Insertion and deletion of keys in B-trees, insertion and search times.

Skip Lists: Need for randomized methods, search and insertion in skip lists, Probabilistic analysis, deterministic skip lists.

Special Types of Binary Trees: AVL trees, Red-Black trees, 2-3 trees, other types.

Module IV: Additional Topics (9L)

Computational Geometry: Introductory concepts, one and two dimensional range searching Priority search trees, priority range trees, quadtrees, k-D trees.

Pattern Matching in Strings: Brute force methods, Boyer-Moore algorithm, Knuth-Morris-Pratt algorithm, Tries, Huffman codes, extensions.

References:

1. T H Cormen, C E Leiserson, R L Rivest, C Stein, Introduction to Algorithms (3rd Ed., 2009), The MIT Press.
2. D E Knuth, The Art of Computer Programming (latest editions), Volume 1 (Fundamental Algorithms) and Volume 3 (Sorting and Searching), Addison Wesley.

Course Name : Research Methodology and IPR					
Course Code: CSEN5102					
Contact Hours Per Week	**L**	**T**	**P**	**Total**	**Credit Points**
	2	**0**	**0**	**2**	**2**

Course Outcomes:

On completion of the course the students undergoing this course are able to:

1. Understand some basic concepts of research and its methodologies
2. Identify appropriate research topics
3. Select and define appropriate research problem and parameters
4. Prepare a project proposal (to undertake a project)
5. Organize and conduct research (advanced project) in a more appropriate manner
6. Write a research report and thesis

Module 1: Introduction:

Definition of Research. Different types of research. Different types of methods for research. Definition of Research Methodology. Research Methods vs. Methodology. Experimental Computer Science versus Theoretical Computer Science.

Module 2:

Part I: Literature Survey and Problem Formulation:

Definition of Literature. Selection of research topic. Survey Procedures. Problem identification. Criteria for prioritizing problems for research. Problem Formulation.

(Discuss in class Web Search: Introduction to Internet. Use of Internet and www. Using of search engines and advanced search tools.)

Part II: Data Collection and Simulation

Module 3: Data Analysis

Analysis tools: Review of Basic Statistical Measures (mean, median, mode, quartile, percentile, variance, covariance, correlation, regression), Probability Distributions (Binomial, Poisson, Uniform, Exponential, Normal), Central Limit Theorem, ANOVA, Latin Square Design, Sampling (Chi-square Distribution, F- Distribution), Test of Hypothesis.

Module 4: Reporting

Technical report writing, Technical paper writing, Plagiarism, Learning Latex

Presentation tool: Introduction to presentation tool, features and functions, creating presentations, customising presentation. [Tools used: Microsoft PowerPoint, Open Office or any other tool]

Spreadsheet tool: Introduction to spread-sheet applications, features and functions, using formulae and functions, data storing, features for statistical data analysis, generating charts/graphs and other features. Functions and Macro [Tools: Microsoft Excel, Open office and similar or other advanced tools]

Patent writing, Patent filing, IPR

References:

1. Research Methodology 2nd Edition, R. Panneerselvam, PHI Publishers.
2. Research Methodology Methods and Techniques, 2nd revised edition, C. R. Kothari, New Age International Publishers.
3. A Guide to LATEX: Document Preparation for Beginners and Advanced Users, 3rd Edition, Helmut Kopka, Patrick W. Daly, Addison-Wesley, 1999.
4. Intellectual Property Rights, Neeraj Pandey, Khushdeep Dharni, PHI Learning Pvt. Ltd., 2014.
5. Microsoft Office Word 2013: A Skills Approach, Inc. Triad Interactive, McGraw-Hill Education, 2014.

Course Name : Advanced Discrete Mathematics and Statistical Methods					
Course Code: MATH5101					
Contact Hours Per Week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

Course Outcomes:

After completing the course the student will be able to:

1. Describe the way of writing mathematical model for real-world optimization problems.
2. Identify Linear Programming Problems and their solution techniques
3. Categorize Transportation and Assignment problems
4. Apply the way in which Game Theoretic Models can be useful to a variety of real-world scenarios in economics and in other areas.
5. Convert practical situations into non-linear programming problems.
6. Solve unconstrained and constrained programming problems using analytical techniques.

MODULE I : Probability and Statistics I (9L)

- Review of Basic Probability: Sample Space, Events, Classical Definition , Addition and Multiplication Rule, Conditional Probability
- Axiomatic definition of Probability and related problems
- Bayes' Theorem and related problems
- Expectation and Variance
- General properties of probability distributions

MODULE II : Probability and Statistics II (9L)

- Moment generating and Characteristic functions
- Special Distributions: Binomial and Normal Distribution
- Measures of Central Tendency: Mean , Median , Mode
- Measures of Dispersion: Standard deviation and Variance
- Relation between random variables, Covariance and Correlation Coefficient
- Linear Regression equations

MODULE-III: Topics in Combinatorial Mathematics (9L)

- Pigeon Hole Principle
- Permutations and Combinations
- Binomial Coefficients
- Recurrence Relations and Generating Functions
- Properties of Fibonacci Numbers
- Principle of Inclusion and Exclusion
- Polya's Theory of Counting, Ramsey's Theorem

MODULE IV: Advanced Graph Theory (9L)

- Tree, Binary Tree, Spanning Tree
- Walk, Path, Cycle, Hamiltonian Graph
- The Travelling Salesman Problem
- Euler Graph, The Chinese Postman Problem , Planar Graph
- Euler's Formula for Planar Graph and Related Problems
- Examples of Non-Planar Graphs, Kuratowski's Theorem
- Matching and Augmenting Paths, Hall's Marriage Theorem and Related Problems
- Vertex Colouring, Chromatic Polynomials

References:

1. Discrete Mathematics and Its Applications, K H Rosen, McGraw Hill
2. Introduction to Graph Theory, D G West, Prentice-Hall of India
3. Discrete Mathematics for Computer Scientists and Engineers, J L Mott, A Kandel and T P Baker, PHI
4. Introduction to Probability and Statistics for Engineers and Scientists, S.Ross, Elsevier
5. Fundamentals of Mathematical Statistics, S.C.Gupta and V.K.Kapoor, Sultan Chand and Sons

CSEN5131 – CSEN5140	Professional Elective I
CSEN5131	Machine Learning
CSEN5132	Advanced Wireless and Mobile Networks
CSEN5133	Introduction to Intelligent Systems
CSEN5134	GPU Computing
CSEN5135	Image Processing

Course Name : Machine Learning					
Course Code: CSEN5131					
Contact Hours Per Week	L	T	P	Total	Credit Points
	3	0	0	3	3

Course Outcomes:

On completion of the course the student should be able to:

1. Learn and understand various machine learning algorithms;
2. Understand complexity of Machine Learning algorithms and their limitations;
3. Compare and contrast various machine learning techniques and to get an insight of when to apply a particular machine learning approach;
4. Mathematically analyze various machine learning approaches and paradigms;
5. Apply common Machine Learning algorithms in practice and implementing their own;
6. Perform experiments in Machine Learning using real-world data

Module 1: Supervised Learning (Regression/Classification) (9L)

- Basic methods: Distance-based methods, Nearest-Neighbors, Decision Trees, Naive Bayes
- Linear models: Linear Regression, Logistic Regression, Generalized Linear Models
- Artificial Neural Networks: Neural Network Model, Back propagation algorithm; Introduction to Radial Basis Function, Recurrent Neural Network, Convolution Neural Network
- Beyond Binary Classification: Multi-class/Structured Outputs, Ranking

Module 2: Unsupervised Learning (9L)

- Clustering: K-means/Kernel K-means, DBScan
- Dimensionality Reduction: PCA and kernel PCA
- Matrix Factorization and Matrix Completion

Module 3: (9L)

- Components of learning; Error and Noise; Training vs Testing: From Training to Testing,
- Dichotomies, Growth Function, Break Points, VC Dimension
- Bias-Variance Tradeoff: Bias and Variance, Learning Curves

Module 4: (9L)

- Support Vector Machines, Nonlinearity and Kernel Methods
- Overfitting: What is overfitting? Dealing with overfitting
- Regularization: informal and formal, Weight decay, Choosing a regularizer
- Deep Learning

References:

1. Kevin Murphy, Machine Learning: A Probabilistic Perspective, MIT Press, 2012
2. Trevor Hastie, Robert Tibshirani, Jerome Friedman, The Elements of Statistical Learning, Springer 2009 (freely available online)
3. Christopher Bishop, Pattern Recognition and Machine Learning, Springer, 2007.
4. Tom Mitchell, Machine Learning, First Edition, McGraw-Hill, 1997.
5. Simon Haykin, Neural Networks and Learning Machines, Third Edition, PHI Learning, 2009.
6. Amit Konar, Computational Intelligence Principles, Techniques and Applications, Springer, 2012.
7. Y. S. Abu-Mostafa, M. Magdon-Ismail, H. T. Lin, Learning from Data - A short Course, AMLbook.com.
8. J. Han and M. Kamber, Data Mining Concepts and Techniques, 3rd, Edition, Morgan Kaufmann Publishers, July 2011.

Course Name : Advanced Wireless and Mobile Networks					
Course Code: CSEN5132					
Contact Hours Per Week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

Course Outcomes:

On completion of the course the student should be able to:

1. Learn the wireless/mobile market and the future needs and challenges.
2. Understand the state-of-the-art in network protocols, architectures and applications.
3. Understand the foundation of understanding and working for future generation of wireless systems
4. Understand the concept of Continuous Time Markov Chain (CTMC)
5. Learn to analyze the quality of a network.
6. Acquire the ability to design new protocols for wireless networks and analyse them.

MODULE 1 (9L)

- INTRODUCTION: Wireless Networking Trends, Key Wireless Physical Layer Concepts, Multiple Access Technologies -CDMA, FDMA, TDMA, Spread Spectrum technologies,
- Frequency reuse, Challenges in Mobile Computing: Resource poorness, Bandwidth, energy etc.
- RADIO PROPAGATION AND MODELLING: Modeling of radio propagation channels including path-loss models, Lognormal shadowing, fading and multipath.
- WIRELESS CELLULAR NETWORKS: 1G and 2G, 2.5G, 3G, Cellular architecture, Frequency reuse, Handoff strategies, Interference and system capacity, Improving coverage and capacity in cellular systems, Spread spectrum Technologies.
- TOOLS TO EVALUATE NETWORK PERFORMANCE: Introduction to Markov Chain, Channel assignment strategies, evaluation of channel assignment strategies using Continuous Time Markov Chain.

MODULE 2 (9L)

- ADVANCED WIRELESS CELLULAR NETWORKS: OFDM, 4G networks, WiMAX (Physical layer, Media access control, Mobility and Networking), LTE
- 5G networks: Network Densification, Millimetre Wave, MIMO
- Convex Optimization and its Application in 5G networks

MODULE 3 (9L)

- NETWORK AND TRANSPORT LAYER PROTOCOLS: Mobile IPv4, Mobile IPv6 and TCP over Wireless Networks: ATCP, ITCP, MTCP and others.
- WLAN: IEEE 802.11 Wireless LANs Physical and MAC layer, 802.11 MAC Modes (DCF and PCF) IEEE 802.11 standards, Architecture and protocols, Infrastructure vs. Adhoc Mode, Hidden Node and Exposed Terminal Problem, Problems, Fading Effects in Indoor and outdoor WLANs, WLAN Deployment issues.
- Cognitive Radio Networks: Analysis of Cognitive Channel Allocation Algorithms using Continuous Time Markov Chain.

MODULE 4 (9L)

- WIRELESS ADHOC NETWORK: Definition, Properties, Limitations, Routing Protocols: DSR, DSDV, AODV, TORA, etc. Introduction to Vehicular Adhoc Networks.
- WIRELESS SENSOR NETWORKS: Introduction, Application, Physical, MAC layer and Network Layer, Power Management, Tiny OS Overview.
- WIRELESS PANs: Bluetooth and Zigbee.
- SECURITY: Security in wireless Networks Vulnerabilities, Security techniques, Wi-Fi Security, DoS in wireless communication.

References:

1. Stallings, William. Wireless communications and networks. Pearson Education India, 2009.
2. Rappaport, Theodore S. Wireless communications: principles and practice. Vol. 2. New Jersey: prentice hall PTR, 1996.
3. Schiller, Jochen H. Mobile communications. Pearson education, 2003.
4. Perkins, Charles E. Ad hoc networking. Vol. 1. Reading: Addison-wesley, 2001.
5. Karl, Holger, and Andreas Willig. Protocols and architectures for wireless sensor networks. John Wiley and Sons, 2007.
6. Boyd, Stephen. Convex optimization. Cambridge university press, 2004.
7. Osseiran, Afif, Jose F. Monserrat, and Patrick Marsch, eds. 5G mobile and wireless communications technology. Cambridge University Press, 2016.

Course Name : Introduction to Intelligent Systems					
Course Code: CSEN5133					
Contact Hours Per Week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

Course Outcomes:

On completion of the course the student should be able to:

1. Understand the basic features / attributes that an intelligent system should have and how those attributes can be incorporated to the system.
2. Comprehend the importance of knowledge as far as intelligence is concerned.
3. Apply this knowledge so that it can be used to infer new knowledge.
4. Apply various searching algorithms as and when required
5. Understand the basic principles of various learning algorithms
6. Design and evaluate the performance of various heuristics in different application domain

Module I: [9L]

- Introduction [1L] – Definition of AI, Intelligent Behavior Turing Test, Typical AI Problems, Various AI Approaches, Limits of AI.
- Introduction to Intelligent Agents [1L] - Agents and environment, Agent Architecture, Agent Performance, Rational Agent, Nature of Environment, Simple Reflex Agent, Goal Based Agent, Utility Based Agent.
- Knowledge Representation and Propositional Logic [2L] - Knowledge representation issues, Approaches to knowledge representation, Propositional Logic – its syntax and semantics, Inference rules, Application of those rules, Limitation of Propositional Logic.
- Problem Solving using Single Agent Search [2L] - Introduction to State-space search, state-space search notation, search problem, Formulation of some classical AI problems as a state space search problem, Explicit Vs. Implicit State space.
- Uninformed Search Techniques [3L] - Basic Principles, Evaluating parameters, BFS, DFS, Depth Limited Search, Iterative Deepening DFS, Uniform Cost Search and Bidirectional Search, Properties of various search methods and their comparative studies.

Module II: [9L]

- Informed Search Methods [5L] - Basic Principles, Heuristics, Best First Search – Greedy Best First, A* Search, their Properties, Admissible and Consistent heuristic, Local Search Techniques – Hill climbing and Simulated Annealing, Comparison with other methods
- Problem Solving using Two Agent Search [2L] - Adversarial Search – Game Tree, MINIMAX Algorithm, Alpha-Beta Pruning, Performance Analysis.
- Constraint Satisfaction Problem [2L] - Definition of CSP, Representation of CSP, Formulation of Various popular problems as CSP, constraint graphs, Solution methods of CSP – Backtracking and Forward Checking, variable and value ordering heuristic, degree heuristic, least-constraining value heuristic, constraint propagation, dependency-directed backtracking

Module III: [9L]

- Knowledge Representation and Predicate Logic [2L] - Syntax and Semantics of FOPL, Representation of facts using FOPL, Clauses, Resolution, Unification methods of inference, Default and Non-Monotonic reasoning.
- Knowledge Representation using Rules [2L] - Rule based system, Horn clauses, Procedural vs. declarative knowledge, forward and backward reasoning, Introduction of logic programming using PROLOG/ LISP.
- Other Representational Formalism [2L] - Inheritable knowledge, Semantic network, Inference in Semantic network, Extending Semantic Network, Frames, Slots as objects.
- Probabilistic reasoning [3L] - Representing knowledge in an uncertain domain, probabilistic inference rules, Naïve Bayes Classifier, Bayesian networks – representation and syntax, semantics of Bayesian net, Fuzzy sets and fuzzy logic.

Module IV: [9L]

- Planning [2L] - Introduction, Simple planning agent, Problem solving vs. planning, Logic based planning, Goal Stack planning, Planning as a search, Total-order vs. partial order planning.
- Learning [5L] - Overview, Taxonomy of learning system, various learning models, learning rules, inductive learning framework, Decision tree based learning, Learning using Neural Network and Genetic Algorithm.

- Natural Language Processing [1L] - Introduction, Brief idea about Syntactic processing, semantic analysis, discourse and pragmatic processing.
- Expert Systems [1L] - Representing and using domain knowledge, expert system shells, knowledge acquisition.

References:

1. Artificial Intelligence A Modern Approach, Stuart Russell and Peter Norvig, Pearson Education
2. Artificial Intelligence, Ritch and Knight, TMH
3. Artificial Intelligence and Intelligent Systems, N.P.Padhy, Oxford University Press
4. Introduction to Artificial Intelligence and Expert Systems, Dan W. Patterson, PHI
5. PROLOG Programming for Artificial Intelligence, Ivan Bratko, Pearson India.

Course Name : GPU Computing					
Course Code: CSEN5134					
Contact Hours Per Week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

Course Outcomes:

By attending the course, students would:

1. Understand GPU architectures to recognize its potential use as general purpose computing unit .
2. Design and implement parallel solution for application kernels using CUDA tools in GPU framework.
3. Conceptualize and apply concurrent data structures to design and analyze efficient parallel algorithms for GPUs by amplifying the utilization of constrained warps, thread blocks, SMP registers, etc.
4. Understand different approaches to handle memory and synchronization issues under parallelism in a GPU framework.
5. Conceptualize the Event-based- Synchronization techniques, used in kernel executions
6. to manage overlapping of data transfers.
7. Understand the application of GPU computing in different graph algorithms and deep learning techniques.

Module1: Introduction (9L)

- A short history of supercomputing;
- GPU Introduction; CPU/GPU comparison; GPU Architecture;
- CUDA Introduction: Hardware overview; OpenCL / OpenACC introduction;

Module 2: CUDA/GPU Programming (8L)

- Grids/Blocks and Threads: Kernels Functions; Thread organization and hierarchy; Warps/ Wavefronts, Thread blocks / Workgroups.
- Memory handling; global, local / shared, private, textures, Constant Memory; Strategy for reducing Global Memory Traffic; Memory and Parallelism;

Module3: Performance Considerations (10L)

- Streaming multiprocessors;1D / 2D / 3D thread mapping; Dynamic Partitioning of Streaming Multiprocessors; Data Prefetching;
- Synchronization: Memory Consistency, Barriers (local versus global), Atomics, Memory fence. Prefix sum, Reduction. Programs for concurrent Data Structures such as Worklists, Linked-lists. Synchronization across CPU and GPU.

Module 4: Advanced Topics (9L)

- Case Studies: Graph algorithms, Deep Learning
- Streams: Asynchronous processing, tasks, Task-dependence, Overlapped data transfers, Default Stream, Synchronization with streams. Events, Event-based- Synchronization - Overlapping data transfer and kernel execution.

Suggested Additional Topics:

- Designing GPU based systems.
- Floating Point considerations in GPU
- Advanced topics: Dynamic parallelism, Unified Virtual Memory, Multi-GPU processing, Peer access, Heterogeneous processing
- Debugging GPU Programs. Profiling,

References:

1. Programming Massively Parallel Processors: A Hands-on Approach; David Kirk, Wen-mei Hwu; Morgan Kaufman;
2. CUDA Programming: A Developer's Guide to Parallel Computing with GPUs; Shane Cook; Morgan Kaufman;
3. GPU Computing and Applications: Yiyu Cai, Simon See; Springer;

Course Name : Image Processing					
Course Code: CSEN5135					
Contact Hours Per Week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

Course Outcomes:

On completion of the course the student should be able to:

1. Get detail exposure to and understanding of various applications of image processing in industry, medicine, and defense.
2. Learn the digital processing algorithms and techniques in image enhancement and image restoration,
3. Able to understand various algorithms used in image compression, segmentation and morphology.
4. Acquire an appreciation for the image processing issues and techniques
5. Apply several image processing techniques in solving real world problems.
6. Conduct independent study and analysis of image processing problems and techniques.

Module I (9L) Fundamentals of Image Processing:

- Image Acquisition, Image Model, Sampling, Quantization, Relationship between pixels, distance measures, connectivity , Image Geometry, Photographic film.
- Histogram: Definition, decision of contrast basing on histogram, operations basing on histograms like image stretching, image sliding, Image classification. Definition and Algorithm of Histogram equalization.

Image Transforms:

- A detail discussion on Fourier Transform,Wavelet transform, DFT,FFT, properties. A brief discussion on WALSH transform, WFT, HADAMARD transform, DCT.

Module II (9L) Image Enhancement: (by SPATIAL Domain Methods):

- Arithmetic and logical operations, pixel or point operations, size operations, Smoothing filters-Mean, Median, Mode filters – Comparative study , Edge enhancement filters – Directorial filters, Sobel, Laplacian, Robert, KIRSCH, Homogeneity and DIFF Filters, prewitt filter, Contrast Based edge enhancement techniques. Low Pass filters, High Pass filters, sharpening filters.

Image enhancement (by FREQUENCY Domain Methods)

- Design of Low pass, High pass, EDGE Enhancement, smoothening filters in Frequency Domain. Butter worth filter, Homomorphic filters in Frequency Domain, advantages of filters in frequency domain, comparative study of filters in frequency domain and spatial domain.

Module III (9L) Image compression

- Definition, A brief discussion on – Run length encoding, contour coding, Huffman code, compression due to change in domain, compression due to quantization , Compression at the time of image transmission. Brief discussion on:- Image Compression standards.

Module IV (9L) Image Segmentation:

- Definition, characteristics of segmentation. Detection of Discontinuities, Thresholding Pixel based segmentation method. Region based segmentation methods – segmentation by pixel aggregation, segmentation by sub region aggregation, histogram based segmentation, spilt and merge technique. Use of motion in segmentation (spatial domain technique only).

Morphology:

- Dilation, Erosion, Opening, closing, Hit-and-Miss transform, Boundary extraction, Region filling, connected components, thinning, Thickening, skeletons , Pruning Extensions to Gray – Scale Images, Application of Morphology in image processing.

Text Books:

1. Digital Image Processing , by Rafael C. Gonzalez and Richard E. Woods Addision Wesley .
2. Digital Image Processing by S. Sridhar, Oxford University Press.

References:

1. Fundamentals of Electronic Image Processing by Arthyr –R – Weeks, Jr. (PHI)
2. Image processing, Analysis, and Machine vision by Milan Sonka vaclan Halavac Roger Boyle, Vikas Publishing House.

CSEN5131 – CSEN5140	Professional Elective II
CSEN5141	Data Science
CSEN5142	Distributed Systems
CSEN5143	Wireless Sensor Networks
CSEN5144	Digital Forensics
CSEN5145	Computational Biology

Course Name : Data Science					
Course Code: CSEN5141					
Contact Hours Per Week	L	T	P	Total	Credit Points
	3	0	0	3	3

Course Outcomes:

On completion of the course the student should be able to:

1. Explain how data is collected, managed and stored for data science;
2. Understand the key concepts in data science, including their real-world applications and some of the popular techniques used by data scientists;
3. Build skills in data management;
4. Demonstrate proficiency with statistical analysis of data;
5. Develop ability to build and assess data-based models;
6. Apply data science concepts and methods to solve real-world problems;

Module 1: Data Science Process (9L)

- Overview – Benefits and Uses, Facets of Data, Big Data Ecosystem;
- Data Scientist's Role in Major Steps – Goal Setting, Data Retrieval, Data Preparation, Data Exploration, Data Modeling, Data Presentation;

Module 2: Statistics for Data Science (9L)

- Exploratory Data Analysis (EDA) – Structured Data, Rectangular Data, Location, Variability, Data Distribution, Binary and Categorical Data, Correlation, Multiple Variables ;
- Sampling Distributions – Random Sampling, Bias, Sampling Distribution and Central Limit Theorem (CLT), Bootstrap, Confidence Interval; Different Distributions –Binomial / Poisson / Exponential / Normal / Student's t / Long-Tailed;

Module 3: Algorithms for Data Science (9L)

- Regression – Simple Linear, Multiple Linear, Polynomial; Prediction using Regression (Logistics Regression); [Non-linear Regression to be covered in detail in ML]
- Classification – K Nearest Neighbors (k-NN), Naive Bayes Classification; Evaluating Classification Models – Accuracy, Precision, Recall, Specificity, Receiver / Operating Characteristics (ROC / OCC) Curve and Area Under Curve (AUC), Precision-Recall (PR) Curve;
- Unsupervised Learning – K-Means Clustering, Hierarchical Clustering and Dendrogram, Principal Component Analysis (PCA); [PCA to be covered in detail in ML]

Module 4: Data Visualization (9L)

- Visual Analytics -- Definition, Roles and Lifecycle / Process / Workflow of Visualization, Common Display Types with Examples;
- Visual Encoding – Guidelines, Mackinlay's Retinal Variables, Mapping Effectiveness; Some Good Practices, Recommendations and Principles related to Graphical Excellence, Some Visualization Techniques;

References:

1. "Introducing Data Science"; Davy Cielen, Arno D Meysman and Mohamed Ali; Dreamtech Press
2. "Practical Statistics for Data Scientists"; Peter Bruce and Andrew Bruce; O'Reilly Media Inc.
3. "Doing Data Science"; Cathy O'Neil and Rachel Schutt; O'Reilly Media Inc.
4. "A First Course in Probability" 8th ed.; Sheldon Ross; Pearson Education Inc.
5. "Mining of Massive Datasets" v2.1; Jure Leskovek, Anand Rajaraman and Jeffrey Ullman; Cambridge University Press

Course Name : Distributed Systems					
Course Code: CSEN5142					
Contact Hours Per Week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

Course Outcomes:

Upon successful completion of this course students should be able to:

1. Identify the introductory distributed database concepts and its structures, and relate the importance and application of emerging database technology
2. Describe terms related to distributed object database design and management.
3. Produce the transaction management and query processing techniques in DDBMS.
4. Demonstrate knowledge of the basic elements and concepts related to distributed system technologies
5. Demonstrate knowledge of the core architectural aspects of distributed systems and underlying components of distributed systems (such as RPC, file systems)
6. Design and implement distributed applications and demonstrate experience in building large-scale distributed applications
7. Use and apply important methods in distributed systems to support scalability and fault tolerance

Module 1:

- Distributed Systems [4L] - Introduction to distributed computing systems. DCS design goals, Transparencies, Fundamental issues
- Distributed Coordination [5L] - Temporal ordering of events, Lamport's logical clocks, Vector clocks; Ordering of messages, Physical clocks, Global state detection

Module 2:

- Process synchronization [5L] - Distributed mutual exclusion algorithms
- Inter-process communication [5L] - Message passing communication, Remote procedure call, Transaction communication, Group communication; Broadcast atomic protocols.

Module 3:

- Distributed Scheduling [5L] - Issues in Load Distributing, Classification of Load Distributing algorithm, Load Balancing vs Load Sharing, Preemptive vs Non-Preemptive transfers
- Distributed file systems [2L] - Introduction, Goal, Architecture, File accessing, sharing, caching, replication.
- Naming [2L] - Design Issues: Naming and Name Resolution, Name Server, Cache Consistency.

Module 4:

- Distributed Databases [8L] - Storage structures for distributed data, data fragmentation, Transparency of distributed architecture, Distributed query processing, and Transaction management in distributed environment, Recovery and Concurrency control, locking protocols, Deadlock handling.

Book:

Text Books:

1. Ceri and Pellagetti: Distributed Database: Principles and Systems, TMH
2. Sukumar Ghosh: Distributed Systems: An Algorithmic Approach, CRC Press
3. Pradeep K Sinha: Distributed Operating Systems Concepts and Design, PHI

Reference:

1. Silberschatz Korth, Sudarshan: Database System Concepts, TMH
2. Connolly and Begg: Database Systems: A practical approach to design, implementation and management, Pearson
3. M. Singhal, N.G. Shivarathri : Advanced Operating Systems, McGraw Hill

Course Name : Wireless Sensor Networks					
Course Code: CSEN5143					
Contact Hours Per Week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

Course Outcomes:

After the completion of this course, students should be able to:

1. Understand the fundamental concepts of wireless sensor networks.
2. Understand the architecture of sensor nodes
3. Acquire basic knowledge and learn the protocols of various layers.
4. Be able to design and implement sensor networks for various application setups.
5. Evaluate the performance of sensor networks and identify bottlenecks
6. Be able to program sensor nodes as per requirement

Module I: Introduction (8L)

- Basic concepts of wireless sensor networks - Motivations, Applications, Performance metrics;
- History and Design factors;
- Architecture of a sensor node;
- Different sensing scenarios using WSN;
- Challenges in implementing WSNs;

Module II: Medium Access Control Protocol design (8L)

- Characteristics of WSN MAC related properties ;
- MAC performance issues ;
- MAC protocols for sensor networks – Schedule based and Random Access based.
- WSN protocols: synchronized, duty-cycled;
- Content based and Contention free MAC protocols;

Module III: Routing protocols for WSN (8L)

- Issues with the adoption of ad hoc routing protocols;
- Data-centric routing ; Position-based / Geographic routing ; Clustering-based routing algorithm, QoS based Routing Protocols.

Module IV: Advanced Topics (12L)

- Security requirements in WSNs ; Different types of attacks in WNs ; Security protocols for WSNs.
- Time Synchronization: Requirements and challenges; Basic Ideas; Various protocols;
- Coverage problem in WSNs: OGDC coverage algorithm ; Placement problem;
- Topology management in WSNs : Different classifications relevant algorithms ; Topology discovery, sleep-cycle management, and clustering;

Suggested Additional Topics:

- Introduction to Markov Chain: Discrete time Markov Chain definition, properties, classification and analysis; MAC Protocol Analysis using Markov chains;
- Sensor Network Programming: Node centric Programming; Macroprogramming; Dynamic reprogramming;
- Introduction to ns-3: Description of the ns-3 core module and simulation example.
- Opportunistic Routing Analysis: Analysis of opportunistic routing (Markov Chain)
- Real Life Deployment; Underwater Sensor Nodes vs Terrestrial Sensor Nodes;
- Power Management.

References:

1. W. Dargie and C. Poellabauer, Fundamentals of Wireless Sensor Networks-Theory and Practice, Wiley 2010.
2. K. Sohraby, D. Minoli and TaiebZnati, “Wireless Sensor Networks -Technology, Protocols, and Applications”, Wiley Interscience 2007.
3. Fei Hu and Xiaojun Cao, “Wireless Sensor Networks: Principles and Practice”, CRC Press, 2010.
4. Takahiro Hara,Vladimir I. Zadorozhny, and Erik Buchmann, “Wireless Sensor Network Technologies for the Information Explosion Era”, Springer 2010.
5. H. Karl and A. Willig, “Protocols And Architectures For Wireless Sensor Networks “, Willey, 2012
6. Q., Muller and Chen, “Security in Wireless Networks and Systems”, Willey, 2011.
7. Stojmenovic, “Wireless Sensor and Actuator Networks: Algorithms and Protocols for Scalable Coordination and Data Communication”, Willey, 2010.

Course Name : Digital Forensics					
Course Code: CSEN5144					
Contact Hours Per Week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

Course Outcomes:

After the completion of this course, students should be able to:

1. Introducing basic concepts of digital forensic science
2. Exploring the specific areas of media, network and code forensics
3. Examining the role of digital forensics in public and private investigations
4. Examining the potential benefits, limitations and risks of digital forensics
5. Increasing awareness of managerial issues raised by the use of digital forensics
6. Enabling students to create disk images, recover deleted files and extract hidden information.
7. Introducing students to the current research in computer forensics. This will encourage them to define research problems and develop effective solutions.

Module 1: Introduction (8L)

- Understanding the Digital Forensics Profession and Investigations; Forensics science, computer forensics, and digital forensics;
- Processing Crime and Incident Scenes; analysis of cyber-criminalistics area, holistic approach to cyber-forensics;

Module 2: Cyber Crime Scene Analysis (10L)

- Understanding Storage Formats for Digital Evidence; Determining the Best Acquisition Method; Using Acquisition Tools: Linux Boot CD; Remote Network Acquisition; Understanding various acquisition tools;
- Evidence Management and Presentation Basics: Identifying Digital Evidence, Rules of Evidence, Understanding Concepts and Terms Used in Warrants;

Module 3: Details of Evidence Management and Presentation (8L)

- Determining the devices to be seized; Getting Location Description; Determining necessary tools; Processing an Incident or Crime Scene; Storing Digital Evidence; Managing Evidence; Obtaining a Digital Hash;
- Sample Criminal Investigation;

Module 4: Digital Forensics Analysis and Validation (10L)

- Approaching Digital Forensics Cases; Analyzing and Validating Forensic Data; Addressing Data-Hiding Techniques; Understanding Steganalysis Methods; Recovering Passwords;
- Network Forensics; Live Acquisition; Using Network Logs and Packet Analyzers;
- E-mail and Social Media Investigations;
- Legal Aspects of Digital Forensics: IT Act 2000, amendment of IT Act 2008.

Suggested Additional Topics:

- Mobile Forensics: mobile forensics techniques, mobile forensics tools.
- Recent trends in mobile forensic technique and methods to search and seizure of electronic evidence
- Working with Windows and CLI Systemsfor File System Recovery
- Understanding Graphic File formats and their recovery.

References:

1. File System Forensic Analysis, by Brian Carrier, Addison-Wesley
2. Handbook of Digital Forensics and Investigation, by Eoghan Casey, Academic Press
3. Guide to Computer Forensics and Investigations 5th Edition, Nelson, Phillips, Steuart, Cengage Learning, 2015
4. The Basics of Digital Forensics, John Sammons, Elsevier
5. Computer Forensics: Computer Crime Scene Investigation, John Vacca, Laxmi Publications
6. Digital Forensic Course Materials from http://mgt2.buffalo.edu/departments/mss/djmurray/mgs610/syllabus.htm

Course Name : Computational Biology					
Course Code: CSEN5145					
Contact Hours Per Week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

Course Outcomes:

1. To become familiar with the use of a wide range of biological databases and their applicability.
2. To understand the storage and retrieval methods of biological data from various biological databases.
3. To study structures of Genes, Molecule codes, DNA Structure.
4. To analyze various existing Graph Algorithms for DNA Sequencing and to compare among different DNA Sequences.
5. To explore different sequenced databases like FASTA, BLAST and evaluate their relevance with research problems.
6. To apply the learned methods to pertinent research problems in various domains.

Module I: (9L)

- Genes, Molecule codes, DNA Structure. DNA and Proteins. Analyzing DNA: copying, cutting and pasting, measuring, probing.
- Exhaustive Search: Restriction Mapping Algorithms, Motif Finding, Finding Median String.

Module II: (9L)

- Greedy Algorithms: Genome Rearrangements, Sorting by Reversals. Greedy approach to Motif Finding.
- Dynamic Programming Algorithms: DNA Sequence Comparison, Edit Distance and Assignments, Longest Common Subsequence, Global Sequence Alignment, Scoring alignments, Local Sequence Alignment, Alignment with Gap Penalties, Multiple Penalties, Gene Prediction, Spliced Alignment.
- Divide and Conquer Algorithms. Sorting, Sequence Alignment, Four-Russians Speedup, Constructing alignments in sub-quadratic time.

Module III: (9L)

- Graph Algorithms: DNA Sequencing, Shortest Superstring Problem, DNA arrays as an alternative sequencing technique. Sequencing by Hybridization: Hamiltonian and Eulerian Path Problems. Protein sequencing and identification. Peptide sequencing problem. Spectrum Graphs: Spectral Convolution, Spectral Alignment.

Module IV: (9L)

- Combinatorial Pattern Matching. Repeat Finding, Exact pattern matching, Keyword trees. Suffix trees. Heuritic similarity search, Approximate pattern matching. Sequenced databases and querying: FASTA, BLAST. Clustering and trees. Gene Expression Analysis, Hierarchical Clustering, Evolutionary trees. Distance based tree reconstruction. Reconstructing trees from additive matrices.
- Evolutionary treesand hierarchical clustering. Charcter based tree reconstruction. Small and Large Parsimony problem.

References:

1. Neil C. Jones and Pavel A. Pevznel: An Introduction to Bioinformatics Algorithms, The MIT Press, 2004.
2. Dan Gusfield: Algorithms on Strings, Trees and Sequences. Cambridge University Press, 1999

Audit Course

DIMA5116	**Disaster Management**
INCO5117	**Constitution of India**
PDLS5118	**Personality Development through Life Enlightenment Skills**
YOGA5119	**Stress Management by Yoga**
SANS5120	**Sanskrit for Technical Knowledge**

Course Name : Disaster Management					
Course Code: CSEN5116					
Contact Hours Per Week	**L**	**T**	**P**	**Total**	**Credit Points**
	2	**0**	**0**	**2**	**0**

Course Outcomes:

After the completion of this course, students should be able to:

1. learn to demonstrate a critical understanding of key concepts in disaster risk reduction and humanitarian response.
2. critically evaluate disaster risk reduction and humanitarian response policy and practice from multiple perspectives.
3. develop an understanding of standards of humanitarian response and practical relevance in specific types of disasters and conflict situations.
4. critically understand the strengths and weaknesses of disaster management approaches, planning and programming in different countries, particularly their home country or the countries they work in.

Module I (6L)

Introduction on Disaster

- Introduction on Disaster - Disaster: Definition; Types of Disaster
- Natural Disaster: such as Flood, Cyclone, Earthquakes, Landslides etc.
- Man-made Disaster: such as Fire, Industrial Pollution, Nuclear Disaster, Biological Disasters, Accidents (Air, Sea, Rail and Road), Structural failures (Building and Bridge), War and Terrorism etc.
- Differences, Nature and Magnitude
- Factors Contributing to Disaster Impact and Severity - Repercussions of various types of Disasters Economic Damage; Loss of Human and Animal Life; Destruction of Ecosystem; Outbreaks of Disease and Epidemics; War and Conflict
- Natural Disaster-prone areas in INDIA - Areas prone to; Earthquake; Floods and Droughts; Landslides and Avalanches; Cyclonic And Coastal Hazards such as Tsunami;
- Lessons Learnt from Recent Disasters

Introduction to Disaster Management

- What is Disaster Management
- Different Phases of Disasters
- Disaster Management Cycles
- Disaster Management Components - Hazard Analysis; Vulnerability Analysis; Prevention and Mitigation; Preparedness; Prediction and Warning; Response; Recovery;
- Disaster Management Act, 2005
- National Disaster Management Structure
- Organizations involved in Disaster Management

Module -II (6L)

Overview on Hazard Analysis and Vulnerability Analysis

Disaster Preparedness

- Disaster Risk Assessment People's Participation in Risk Assessment

- Disaster Risk Reduction
- Preparedness Plans
- Community preparedness: Emergency Exercises/ Trainings/Mock Drills

Disaster Prediction and Warning

- Activities - Tracking of disaster; Warning mechanisms; Organizational response; Public education; Communication; Evacuation planning
- Current tools and models used for prediction and early Warnings of disaster - Application of Remote Sensing; Data From Meteorological and other agencies; Smartphone/ Web based Apps for Disaster Preparedness and Early Warning used in different parts of Globe

Module -III (6L)

Disaster Response

- Crisis Management: The Four Emotional Stages of Disaster - Heroic Phase; Honeymoon Phase; Disillusionment Phase; Reconstruction Phase
- Need for Coordinated Disaster Response - Search, Rescue, Evacuation, Medical Response and Logistic Management; Psychological Response and Management (Trauma, Stress, Rumor and Panic)
- Role of Government, International and NGO Bodies

Post-disaster Situation Awareness

- Need for Situation Awareness in Post Disaster scenario
- Challenges in communication of situational data from affected areas
- Need for community-driven disaster management for reliable situation awareness
- Crowd-sourcing of situational data: Issues and challenges

Post-disaster Damage and Need Assessment

- Current Trends and Practices – RAPID Damage and Need Assessment
- SPHERE standards in Disaster Response
- ICT based techniques for Post-disaster damage and need assessment

Module -III (6L)

Rehabilitation, Reconstructions and Recovery

- Reconstruction and Rehabilitation as a Means of Development.
- Post Disaster effects and Remedial Measures
- Creation of Long-term Job Opportunities and
- Livelihood Options
- Disaster Resistant House Construction
- Sanitation and Hygiene
- Education and Awareness
- Dealing with Victims' Psychology
- Long-term Counter Disaster Planning

Disaster Mitigation

- Meaning, Concept and Strategies of Disaster Mitigation
- Emerging Trends in Mitigation
- Structural Mitigation and Non-Structural Mitigation
- Programs of Disaster Mitigation In India

References:

1. R. Nishith, Singh AK, "Disaster Management in India: Perspectives, issues and strategies", New Royal book Company.
2. Sahni, Pardeep et.al. (Eds.)," Disaster Mitigation Experiences And Reflections", Prentice Hall of India, New Delhi.
3. Goel S. L., Disaster Administration And Management Text And Case Studies", Deep and Deep Publication Pvt. Ltd., New Delhi.

Course Name : Constitution of India					
Course Code: INCO5117					
Contact Hours Per Week	**L**	**T**	**P**	**Total**	**Credit Points**
	2	**0**	**0**	**2**	**0**

Course Objectives

1. Understand the premises informing the twin themes of liberty and freedom from a civil rights perspective.
2. To address the growth of Indian opinion regarding modern Indian intellectuals' constitutional role and entitlement to civil and economic rights as well as the emergence of nationhood in the early years of Indian nationalism.
3. To address the role of socialism in India after the commencement of the Bolshevik Revolution in 1917 and its impact on the initial drafting of the Indian Constitution

Course Outcomes:

After the completion of this course, students should be able to:

1. Discuss the growth of the demand for civil rights in India for the bulk of Indians before the arrival of Gandhi in Indian politics.
2. Discuss the intellectual origins of the framework of argument that informed the conceptualization of social reforms leading to revolution in India.
3. Discuss the circumstances surrounding the foundation of the Congress Socialist Party [CSP] under the leadership of Jawaharlal Nehru and the eventual failure of the proposal of direct elections through adult suffrage in the Indian Constitution.
4. Discuss the passage of the Hindu Code Bill of 1956.

Module I (8L)

- History of Making of the Indian Constitution: History, Drafting Committee, (Composition and Working)
- Philosophy of the Indian Constitution: Preamble, Salient Features

Module II (4L)

- Contours of Constitutional Rights and Duties: Fundamental Rights, Right to Equality, Right to Freedom, Right against Exploitation, Right to Freedom of Religion, Cultural and Educational Rights, Right to Constitutional Remedies, Directive Principles of State Policy, Fundamental Duties.

Module III (4L)

- Organs of Governance: Parliament, Composition, Qualifications and Disqualifications, Powers and Functions, Executive, President, Governor, Council of Ministers, Judiciary, Appointment and Transfer of Judges, Qualifications, Powers and Functions

Module IV (8L)

- Local Administration: District's Administration head: Role and Importance; Municipalities: Introduction, Mayor and role of Elected Representative, CEO of Municipal Corporation; Pachayati raj: Introduction, PRI: ZilaPachayat; Elected officials and their roles; CEO ZilaPachayat: Position and role; Block level: Organizational Hierarchy (Different departments); Village level: Role of Elected and Appointed officials, Importance of grass root democracy
- Election Commission: Election Commission: Role and Functioning; Chief Election Commissioner and Election Commissioners; State Election Commission: Role and Functioning; Institute and Bodies for the welfare of SC/ST/OBC and women.

References

1. The Constitution of India, 1950 (Bare Act), Government Publication.
2. Dr. S. N. Busi, Dr. B. R. Ambedkar framing of Indian Constitution, 1st Edition, 2015.
3. M. P. Jain, Indian Constitution Law, 7th Edn., Lexis Nexis, 2014.
4. D.D. Basu, Introduction to the Constitution of India, Lexis Nexis, 2015.

Course Name : Personality Development through Life Enlightenment Skills					
Course Code: PDLS5118					
Contact Hours Per Week	**L**	**T**	**P**	**Total**	**Credit Points**
	2	**0**	**0**	**2**	**0**

Course Outcomes:

After the completion of this course, students should be able to:

1. Study of Shrimad-Bhagwad-Geeta will help the student in developing his personality and achieve the highest goal in life
2. The person who has studied Geeta will lead the nation and mankind to peace and prosperity
3. Study of Neetishatakam will help in developing versatile personality of students.

Module I (6L)

Neetisatakam-Holistic development of personality

- Verses- 19,20,21,22 (wisdom)
- Verses- 29,31,32 (pride and heroism)
- Verses- 26,28,63,65 (virtue)

Module II (6L)

Approach to day to day work and duties.

- Verses- 52,53,59 (dont's)
- Verses- 71,73,75,78 (do's)
- Shrimad Bhagwad Geeta : Chapter 2-Verses 41, 47,48,

Module III (6L)

Statements of basic knowledge.

- Chapter 3-Verses 13, 21, 27, 35, Chapter 6-Verses 5,13,17, 23, 35,
- Chapter 18-Verses 45, 46, 48.
- Shrimad Bhagwad Geeta: Chapter2-Verses 56, 62, 68
- Chapter 12 -Verses 13, 14, 15, 16,17, 18

Module IV (6L)

Personality of Role model.

- Shrimad Bhagwad Geeta: Chapter2-Verses 17, Chapter 3-Verses 36,37,42,
- Chapter 4-Verses 18, 38,39
- Chapter18 – Verses 37,38,63

References

1. "Srimad Bhagavad Gita" by Swami Swarupananda Advaita Ashram (Publication 2. Department), Kolkata
2. Bhartrihari's Three Satakam (Niti-sringar-vairagya) by P.Gopinath, Rashtriya Sanskrit Sansthanam, New Delhi.

Course Name : Stress Management by Yoga					
Course Code: YOGA5119					
Contact Hours Per Week	**L**	**T**	**P**	**Total**	**Credit Points**
	2	**0**	**0**	**2**	**0**

Course Objectives

4. To achieve overall health of body and mind
5. To overcome stress

Course Outcomes:

After the completion of this course, students should be able to:

5. Develop healthy mind in a healthy body thus improving social health also
6. Improve efficiency

Module I (6L)

- Definitions of Eight parts of yog. (Ashtanga)

Module II (6L)

Yam and Niyam.
Do`s and Don't's in life.

- Ahinsa, satya, astheya, bramhacharya and aparigraha
- Shaucha, santosh, tapa, swadhyay, ishwarpranidhan

Module III (6L)

Asan and Pranayam

- Various yog poses and their benefits for mind and body

Module IV (6L)

- Regularization of breathing techniques and its effects-Types of pranayam

References

1. 'Yogic Asanas for Group Tarining-Part-I" :Janardan Swami Yogabhyasi Mandal, Nagpur
2. "Rajayoga or conquering the Internal Nature" by Swami Vivekananda, AdvaitaAshrama (Publication Department), Kolkata

Course Name : Sanskrit for Technical Knowledge					
Course Code: SANS5120					
Contact Hours Per Week	**L**	**T**	**P**	**Total**	**Credit Points**
	2	**0**	**0**	**2**	**0**

Course Objectives

1. To get a working knowledge in illustrious Sanskrit, the scientific language in the world
2. Learning of Sanskrit to improve brain functioning
3. Learning of Sanskrit to develop the logic in mathematics, science and other subjects
4. Enhancing the memory power
5. The engineering scholars equipped with Sanskrit will be able to explore the
6. Huge knowledge from ancient literature

Course Outcomes:

After the completion of this course, students should be able to:

1. Understanding basic Sanskrit language
2. Ancient Sanskrit literature about science and technology can be understood
3. Being a logical language will help to develop logic in students

Module I (6L)

- Alphabets in Sanskrit,
- Past/Present/Future Tense,

Module II (6L)

- Simple Sentences
- Order

Module III (6L)

- Introduction of roots
- Technical information about Sanskrit Literature

Module IV (6L)

- Technical concepts of Engineering-Electrical, Mechanical, Architecture, Mathematics

References

1. "Abhyaspustakam" – Dr.Vishwas, Samskrita-Bharti Publication, New Delhi
2. "Teach Yourself Sanskrit" Prathama Deeksha-VempatiKutumbshastri, Rashtriya Sanskrit Sansthanam, New Delhi Publication
3. "India's Glorious Scientific Tradition" Suresh Soni, Ocean books (P) Ltd., New Delhi.

Course Name : Advanced Data Structures Lab					
Course Code: CSEN5151					
Contact Hours Per Week	**L**	**T**	**P**	**Total**	**Credit Points**
	0	**0**	**4**	**4**	**2**

Course Outcomes:

At the end of this lab session, the student will

1. be able to design and analyze the time efficiency of various data structures
2. be capable to identity the appropriate data structure for a given problem
3. be able to write program with appropriate data structures for a given problem
4. have practical knowledge on the applications of data structures

A tentative outline for this laboratory is given below:

- Assignments on the application of array data structure to sort a set of elements using different sorting methods (e.g., bubble sort, insertion sort, selection sort)
- Assignments on the application of array data structure to search an element in a set of elements using different searching methods (e.g., linear search, binary search)
- Implementation of stack and queue using array and linked list data structures
- Assignments on the implementation of binary tree using array and linked list and traversal of the tree
- Implementation of hashing where collision resolution is done using open addressing method
- Implementation of KMP algorithm for pattern matching
- Application of one/ two data structures in real life applications

Course Name : Machine Learning Lab					
Course Code: CSEN5181					
Contact Hours Per Week	**L**	**T**	**P**	**Total**	**Credit Points**
	0	**0**	**4**	**4**	**2**

Course Outcomes:

At the end of this lab session, the student will be able to

1. write code the machine learning algorithm in C or Python.
2. understand and conceptualize the methods of machine learning and its applications.
3. design simple algorithms for pattern classification, code them with Python programming language and test them with benchmark data sets.
4. write program analyze and evaluate simple algorithms for pattern classification.
5. analyze and evaluate simple algorithms of estimation.
6. design complex machine learning algorithms using tools like Excel, R, TensorFlow, Weka.

List of Experiments:

- Regression (single and Multiple Variables) linear and non-liner;
- Logistic regression
- Classifiers - K-NN; Naïve Bayes Classifier; Perceptron; Multi Layer Perceptron.
- Clustering Algorithms - K-Means; DB-Scan
- Familiarization with a few ML Tools Excel; WEKA; R; Python; TensorFlow
- Applications of ANN and SVM using ML tools

Course Name : Advanced Wireless and Mobile Networks Lab					
Course Code: CSEN5182					
Contact Hours Per Week	**L**	**T**	**P**	**Total**	**Credit Points**
	0	**0**	**4**	**4**	**2**

Course Outcomes:

1. The students should get familiar with the various network simulators like ns2 and QualNet.
2. To learn to model and simulate various network topologies
3. To learn how to evaluate MAC and network protocols using network simulation software tools.
4. To learn the methodology to develop new MAC and network protocols and simulate them in the network simulators.

Syllabus:

- Network Simulator (NS)
  - Installation of Network Simulator ns 2
  - Familiarization with ns 2
  - Learn programming in OTCL
  - Setup wired and wireless networks using existing protocols in OTCL
  - Observe the variation in the network performance of wireless ad hoc networks for various routing protocols
  - Observe the variation in the network performance of vehicular ad hoc networks for various routing protocols
- Real time network simulator Qualnet
  - Familiarization with the real time network simulator Qualnet.
  - Learn to setup wired and wireless networks, add applications, run scenarios, obtain results and analyze them.
  - Observe the variation in the network performance of wireless ad hoc networks forvarious routing protocols.
  - Observe the variation in the network performance of vehicular ad hoc networks for various routing protocols.

Course Name : Introduction to Intelligent Systems Lab					
Course Code: CSEN5183					
Contact Hours Per Week	**L**	**T**	**P**	**Total**	**Credit Points**
	0	**0**	**4**	**4**	**2**

Course Outcomes:

At the end of this course, students are expected to

1. get a good flavor of logical programming by using PROLOG/ LISP
2. apply that knowledge to solve some intelligent puzzles
3. learn to model simple intelligent systems
4. apply search algorithm in developing intelligent systems

In this laboratory students will be familiarized with PROLOG/ LISP language.

- Introduction to PROLOG facts and rules with the help of a simple family tree: how the goals are given in PROLOG; some simple queries on the family tree formation of recursive definition; how PROLOG executes the goals; simple assignments
- how PROLOG deals with problems with numbers – integers, real; with some examples
  Introduction to LIST structure: how PROLOG implements LIST; some simple assignments on LIST
- some more complex assignments on LISTIntroduction to Accumulator – simple assignments
- Introduction to CUT with simple assignments; implementation of sorting algorithms
- implementation of graph search algorithms like DFS, BFS; Some application of BFS. DFS
- Implementation of some well known puzzles, like 8-queens problem, Towers-of-Hanoi problem, Missionaries and Cannibals problem etc.
- Introduction to LISP
- Some simple assignments on LISP

References:

1. PROLOG Programming for Artificial Intelligence, Ivan Bratko, Pearson India.

Course Name : GPU Computing Lab					
Course Code: CSEN5184					
Contact Hours Per Week	**L**	**T**	**P**	**Total**	**Credit Points**
	0	**0**	**4**	**4**	**2**

Course Outcomes:

By attending the course, students would:

1. learn concepts in parallel programming,
2. gain the knowledge of implementation of programs on GPUs,
3. have skills for debugging and profiling parallel programs on CUDA.

Experiments will be conducted under Linux on any (say, ARCUS) GPU cluster. The header files (helper_cuda.h, helper_string.h) which come from the CUDA SDK will be used. They provide routines for error-checking and initialization.

- **Lab 1: Application**: a trivial "hello world" example
- **CUDA aspects**: launching a kernel, copying data to/from the graphics card, error checking and printing from kernel code
- **Lab 2: Application**: Monte Carlo simulation using NVIDIA's CURAND library for random number generation
- **CUDA aspects**: constant memory, random number generation, kernel timing, minimising device memory bandwidth requirements
- **Lab 3: Application**: reduction
- **CUDA aspects**: dynamic shared memory, thread synchronization;
- **Lab 4**: Matrix-matrix operation (CUDA)
- **Lab 5**: Application: using the CUBLAS and CUFFT libraries
- **Lab 6**: Matrix-matrix operation via cuBLAS (CUDA)
- **Lab 7: Application**: tri-diagonal equations
- **Lab 8**: pattern matching
- **Lab 9 and Lab 10**: streams and OpenMP multithreading:

References:

1. Programming Massively Parallel Processors: A Hands-on Approach; David Kirk, Wen-mei Hwu; Morgan Kaufman.
2. CUDA Programming: A Developer's Guide to Parallel Computing with GPUs; Shane Cook; Morgan Kaufman
3. GPU Computing and Applications: Yiyu Cai, Simon See; Springer.
4. Web Limk: https://people.maths.ox.ac.uk/gilesm/cuda/

Course Name : Image Processing Lab					
Course Code: CSEN5185					
Contact Hours Per Week	**L**	**T**	**P**	**Total**	**Credit Points**
	0	**0**	**4**	**4**	**2**

Course Outcomes:

1. Students will learn to convert one image form to another image form.
2. Able to learn various kinds of image enhancement and image restoration techniques.
3. They will learn various techniques of image compression, image segmentation etc.

List of Experiments:

- Display of Grayscale Images.
- Histogram Equalization.
- Non-linear Filtering.
- Edge detection using Operators.
- 2-D DFT and DCT.
- Filtering in frequency domain.
- Filtering in spatial domain.
- Display of color images.
- DWT of images.
- Segmentation using watershed transform.
- Image Compression.
- Applications of image zooming and image shrinking etc.

M. Tech. Detailed Syllabus - Semester II

Subject Name: Advanced Algorithms					
Paper Code: CSEN5201					
Contact Hours per week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

Course Outcomes:

After completion of the course, students would be able to:

1. Remember time complexities of various existing algorithms in different situations
2. Understand the basic principles of different paradigms of designing algorithms
3. Apply mathematical principles to solve various problems
4. Analyze the complexities of various algorithms
5. Evaluate the performance of various algorithms in best case, worst case and average case
6. Create/ Design a good algorithm for a new problem given to him/ her

Module I [9L]

- **Basic Concepts [3L]:** Review of basic data structures and algorithms, worst-case and average-case analyses, asymptotic complexity, Big-O, Big-Theta, Big-Omega and small-o notations and their properties, introduction to recurrences, suitable examples.
- **Sorting and Selection [4L] :** merge sort, quick sort, heap sort and their analysis; priority queues, lower bounds for comparison-based sorting, median and order statistics, selection of *k*-th largest element.
- **Searching [2L]:** Linear Search, Binary Search, Analysis in best case, worst case and average case.

Module II [9L]

- **Graph Algorithms [3L] :** Graph traversal algorithms: BFS and DFS; topological sorting of cycle-free graphs, strongly connected components.
- **Greedy Method [6L]:** Elements of the greedy strategy, fractional knapsack problem; Shortest Path Algorithms: Dijkstra's and Bellman Ford with correctness proofs; Minimum cost spanning trees: Prim's and Kruskal's algorithms and their correctness proofs.

Module III [9L]

- **Dynamic Programming [4L]:** Basic Principles, Matrix-chain multiplication algorithm, All pairs shortest path algorithm - Floyd-Warshall algorithm, LCS Problem; Some more problems.
- **Algebraic Operations [2L]:** Integer multiplication, GCD computation using Euclid's algorithm, polynomial evaluation, Strassen's matrix multiplication algorithm.
- **Amortized Analysis [3L]:** Aggregate, Accounting and Potential Methods, Example problems.

Module IV [9L]

- **Flows in Networks [2L]:** Basic Concepts, maxflow – mincut theorem, Ford-Fulkerson method, Edmond-Karp maximum-flow algorithm.
- **NP-completeness [3L]:** Informal concepts of deterministic and non-deterministic algorithms, P and NP, NP-completeness, Cook's theorem, examples of NP-complete problems.
- **Approximation algorithms [3L]:** Necessity of approximation schemes, performance guarantee, Approximation algorithms for 0/1 knapsack, vertex cover, TSP.
- **Recent Trends [1L]:** Discussion on recent searching and sorting techniques by applying recently proposed data structures.

References:

1. T.H.Cormen, C.E.Leiserson, R.L.Rivest, C Stein: Introduction to Algorithms (2nd Ed), MIT Press, 2001.
2. G Brassard, P Bratley: Introduction to Algorithmics, Pearson Prentice Hall, 1996
3. D. E. Knuth: The Art of Computer Programming (2nd Ed or later), vol 1-3, Addison-Wesley
4. J Kleinberg, E Tardos: Algorithm Design, Pearson, 2006

Subject Name: Soft Computing					
Paper Code: CSEN5202					
Contact Hours per week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

Course Outcomes:

After completion of course, students would be able to:

1. Learn (remember) and understand soft computing techniques and their roles in building intelligent machines.
2. Apply fuzzy logic and reasoning to handle uncertainty and solve various engineering problems.
3. Design (create) methodology to solve optimization problems using genetic algorithms
4. Analyze and evaluate solutions by various soft computing approaches for a given problem.
5. Understand various models of artificial neural networks and their applications in solving pattern recognition and machine learning problems.
6. Develop intelligent systems leveraging the paradigm of soft computing techniques.

Module I: (L9): Introduction to Soft Computing and Fuzzy Logic

Evolution of Computing, Soft Computing Constituents. Fuzzy Sets, Operations on Fuzzy Sets, and Membership Functions, Fuzzy Relations, Fuzzy Rules and Fuzzy Reasoning, Fuzzy Inferences, Fuzzy logic controller design.

Module II: (L9): Neural Networks

Introduction to Artificial Neural Network (ANN). Different ANN architectures, Traing techniques for ANNs (Supervised Learning Neural Networks, Radial Basis Function Networks, Unsupervised Learning Neural Networks, Self organizing map).

Module III (L9): Genetic Algorithms

Introduction to Genetic Algorithms (GA), Binary coded GA, Real coded GA, Other coding methods, Non-dominated Sorted GA, Solving single-objective optimization problems using GAs.

Module IV (L9): Applications and Advanced Soft Computing tools

Concept of multi-objective optimization problems (MOOPs) and issues of solving them, GAs to solve MOOPs, neural network in deep learning, Neuro-Fuzzy Hybrid system.

References:

1. George J. Klir and Bo Yuan, Fuzzy Sets and Fuzzy Logic:Theory and Applications, Prentice Hall, 1995.
2. Fuzzy Logic with Engineering Applications (3rd Edn.), Timothy J. Ross, Willey, 2010
3. Fuzzy Logic for Embedded Systems Applications, Ahmed M. Ibrahim, Elesvier Press, 2004.
4. An Introduction to Genetic Algorithms, Melanie Mitchell, MIT Press, 2000
5. Genetic Algorithms In Search, Optimization And Machine Learning, David E. Goldberg, Pearson Education, 2002
6. Neural Networks, Fuzzy Logis and Genetic Algorithms : Synthesis, and Applications, S. Rajasekaran, and G. A. Vijayalakshmi Pai, Prentice Hall of India, 2007.
7. Soft Computing, D. K. Pratihar, Narosa, 2008.
8. Neural Networks and Learning Machines, (3rd Edn.), Simon Haykin, PHI Learning, 2011.

Professional Elective III

CSEN5231 – CSEN5235	Professional Elective III
CSEN5231	Data Preprocessing and Analysis
CSEN5232	Secure Software Design and Enterprise Computing
CSEN5233	Computer Vision
CSEN5234	Theory of Computation
CSEN5235	Computational Geometry

Subject Name: Data Preprocessing and Analysis					
Paper Code: CSEN5231					
Contact Hours per week	L	T	P	Total	Credit Points
	3	0	0	3	3

Course Outcome:

After completion of course, students would be able to

1. Acquire knowledge in a broad range of methods based on statistics and informatics for data preprocessing and analysis and tools for visualizing the main characteristics of data.
2. Understand the whole process line of gathering relevant data, preprocessing the data, performing exploratory analysis on the data and visualizing the implicit knowledge extracted from data.
3. Apply suitable methods for unveiling the underlying structure of the data, testing underlying assumptions in various fields.
4. Analyze the results of experiment with the help of various visualization tools and statistical tests.
5. Evaluate the performance of not only a computational method after obtaining different results by using different parameter values in order to choose the correct parameter value, but also, all similar methods in order to find out the best performing algorithm for a dataset.
6. Get familiar with relevant literatures, derive theoretical properties of the existing methods and come up with novel approach or pipeline for analyzing data across various fields by solving assignment problems.

Module I (7L): Data Gathering and Preparation

Data formats- Structured, semi-structured and unstructured data format; **Parsing and transformation-** Need of Parsing, Text markup language and parser, Extensible mark-up language and parser; **Scalability-** Goals of a scalable platform, when to scale your database, vertical scaling, read scaling; **Real-time issues-** Real-time event transfer, Real-time situation discovery, Real-time analytics, Real-time decision making, Real-time responses.

Module II (7L): Data Cleaning

Importance of data cleaning, Data quality dimensions (Accuracy, Completeness, Currency and Consistency); **Classification of Data quality problems (Single-source and Multi-source problems)-** Consistency checking, Heterogeneous and missing data; **Data Cleaning Approaches:** Data Transformation and segmentation.

Module III (15L): Exploratory Analysis

Descriptive statistics- Central Tendency, variation, shape; **Comparative statistics-** Parametric (Paired t-test, Unpaired t-test, Repeated measures ANOVA, One-way ANOVA, Pearson correlation) vs non-parametric tests (Wilcoxon- Signed Ranks test, Mann-Whitney test, Friedman test, Kruskal-Wallis test, Spearman correlation, Chi-Square test/ Fisher's test); **Clustering and association-** Overview of clustering, distance metrics, k-means and hierarchical clustering; **Hypothesis generation-** Introduction, Null hypothesis, alternate hypothesis, Types of errors. Multiple testing- Methods for addressing multiple testing (Family wise error rate and False discovery rate).

Module IV (7L): Visualization

Designing visualizations- Steps in designing visualization, Problems in Designing Effective Visualization **Time series-** Line Graph, Stacked Area Chart, Bar Chart, Gantt Chart, Stream Graph, Heat Map, Polar Area Diagram; **Geolocated data-** Introduction spatial data, Point phenomena, line phenomena, area phenomena, Cartograms; **Correlations and connections-** Marimekko chart, Parallel Coordinates plot, Radar chart, Venn diagram, bubble chart, heatmap, scatter plot, arc diagram, brainstorm, chord diagram, connection map, network diagram, non-ribbon chord diagram, tree diagram; **Hierarchies and networks-** Space and Non-space filling methods, Node-link graphs, Matrix representation of graphs; **Interactivity-** Interaction operators, Interaction operands and spaces.

References:

1. Making sense of Data: A practical Guide to Exploratory Data Analysis and Data Mining, by Glenn J. Myatt, Wiley Interscience.
2. Data Quality: Concepts, Methodologies and Techniques, by Carlo batini and Monica Scannapieca, Springer.
3. Fundamentals of Descriptive Statistics, by Zealure C. Holcomb, Routledge.
4. Visualizing Data: Exploring and Explaining Data with the Processing Environment, by Ben Fry, O'REILLY'.

Subject Name: Secure Software Design and Enterprise Computing					
Paper Code: CSEN5232					
Contact Hours per week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

COURSE OUTCOMES:

After completion of course, students would be able to

1. Understand methodologies and tools to design and develop secure software containing minimum vulnerabilities and flaws.
2. Study various issues like weak random number generation, information leakage, poor usability, and weak or no encryption on data traffic.
3. Know essential techniques for reducing and avoiding system and software security problems,
4. Evaluate various enterprise application design and development tools and standard practices.
5. Review techniques for successfully implementing and supporting network services on an enterprise scale and heterogeneous systems environment.
6. Solve enterprise scale problems emanating from lapses in security requirements and information system management practices.

Module 1:

Secure Software Design **(10L)**

Identify software vulnerabilities and perform software security analysis;
Exposure to security programming practices;
Basics of fundamental software security design concepts;
Perform security testing and quality assurance.
Domain Model for Security Risk Management;
Security Risk; Security Requirements and Metrics;
Security Modeling: Understanding security goals and business activities;
Designing secure system functions and behavior; Role-based access control ;

Module 2: **(8L)**

Enterprise Application Development

Describe the nature and scope of enterprise software applications;
Explore technologies available for the presentation, business and data tiers of an enterprise software application;
Design and build a database using an enterprise database system;
Develop components at the different tiers in an enterprise system; Design and develop a multi-tier solution to a problem using technologies used in enterprise system.

Module 3: **(8L)**

Enterprise Systems Administration

Design, implement and maintain a directory-based server infrastructure in a heterogeneous systems environment;
Monitor server resource utilization for system reliability and availability;
Install and administer network services (DNS/DHCP/Terminal Services/Clustering/Web/Email).

Module 4: **(10L)**

Enterprise Network Infrastructure

Obtain the ability to manage and troubleshoot a network running multiple services, Understand the requirements of an enterprise network and how to go about managing them.
Handle insecure exceptions and command/SQL injection, Defend web and mobile applications against attackers, software containing minimum vulnerabilities and flaws.
Case study: DNS server, DHCP configuration and SQL injection attack.

References:

1. Theodor Richardson, Charles N Thies, Secure Software Design, Jones and Bartlett
2. Kenneth R. van Wyk, Mark G. Graff, Dan S. Peters, Diana L. Burley, Enterprise Software Security, Addison Wesley.
3. Principles of Secure Software Design: Dr. Raimundas Matuleviičius
4. Architecting Applications for the Enterprise: Dino Esposito and Andrea Saltarello; Microsoft Press;
5. Enterprise Applications Administration: Jeremy Faircloth; Morgan Kaufmann publishers;
6. RedHat Linux Networking and System Administration: Terry Collings and Kurt Wall; Wiley Publishing;
7. SQL Injection Attacks and Defense: Justin Clarke; Elsevier Publishing;

Subject Name: Computer Vision					
Paper Code: CSEN5233					
Contact Hours per week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

COURSE OUTCOMES:

After completion of course, students would be able to:

1. Learn basic concepts, terminology, theories, models and methods in the field of image analysis and computer vision.
2. Learn and understand shape and region analysis.
3. Apply the vision technology in solving image processing and computer vision problems.
4. Identify the limitations of vision systems.
5. Develop skills to implement boundary detection and motion related techniques.
6. Design successful applications to process and analyze images.

Module I (8L):

Overview, computer imaging systems, lenses, Image formation and sensing, Image analysis, pre-processing and Binary image analysis, Edge detection, Edge detection performance, Hough transform, corner detection.

Module II (7L):

Fourier Transform, Segmentation, Morphological filtering.

Module III (10L):

Feature extraction, shape, histogram, color, spectral, texture, using CVIP tools, Feature analysis, feature vectors, distance /similarity measures, data preprocessing.

Module IV (11L):

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

Recent trends in Activity Recognition, Computational photography, Biometrics.

References:

1. Computer Vision: Algorithms and Applications by Richard Szeliski.
2. Deep Learning, by Goodfellow, Bengio, and Courville.
3. Dictionary of Computer Vision and Image Processing, by Fisher et al.

Subject Name: Theory of Computation					
Paper Code: CSEN5234					
Contact Hours per week	L	T	P	Total	Credit Points
	3	0	0	3	3

Course Outcomes:

Students who complete the course will demonstrate the ability to do the followings:

1. Design and analyze Deterministic and non-deterministic finite state automata.
2. Understand the correspondence between finite state automata and regular languages.
3. Design context free grammars to generate strings from a context free language and convert them into Chomsky normal forms.
4. Design deterministic and non-deterministic push down automata to recognize context free languages.
5. Construct Touring machines for computable functions.
6. Understand the hierarchy of formal languages, grammars and machines.
7. Distinguish between computability and non-computability and Decidability and undecidability.

Module 1: (9 hours)

Finite State Machines. Basic definitions, state transition diagrams, state tables, Mealy model, Moore model, formal mathematical definition, input alphabet, input strings, concept of language. Recognition of a language by a finite state automaton. Examples of design of FSMs.

Distinction between deterministic and non-deterministic automaton, conversion of a non-deterministic machine to deterministic form. Epsilon transitions and their elimination.

Regular grammars and languages.

Module 2: (8 hours)

Regular Expressions. Definition and properties of regular expressions. Correspondence between regular expressions and finite state machines. Kleene's Theorem.

Types of Languages. The Pumping Lemma for Type 3 languages. Examples of languages that are not regular. Closure properties of regular languages. Decision properties of regular languages.Capabilities and limitations of FSMs.Applications of finite automata.

Module 3: (7 hours)

Context-free grammars. Parse Trees. Pushdown Automata. Deterministic and non-deterministic pushdown automata. Designing PDAs to accept Type 2 languages. Chomsky Normal Form of context-free grammars. Ambiguity of CFLs. Examples of ambiguous grammars.

Pumping lemma for CFLs. Examples of languages that are not context-free. Closure properties of CFLs. Decision properties of CFLs.

Module 4: (12 hours)

Turing machines: Church's Thesis. The Turing Machine model. Design of TMs. Variants of the basic TM model. Non-deterministic TMs.

Undecidability. Recursive and recursively enumerable languages. Examples of undecidable problems. Post's Correspondence Problem.

Introduction to Computational Complexity Theory: Intractable Problems. P and NP. NP complete problems. Examples of polynomial-time reductions.

References

1. Michael Sipser, Introduction to the Theory of Computation (3rd ed), PWS Publishing.
2. John E. Hopcroft, Rajeev Motwani and Jeffrey D. Ullman, Introduction to Automata Theory, Languages, and Computation, Pearson Education Asia.
3. Peter Linz, An Introduction to Formal Languages and Automata (6th ed), Jones and Bartlett Learning.
4. Harry R. Lewis and Christos H. Papadimitriou, Elements of the Theory of Computation, Pearson Education Asia.

Subject Name: Computational Geometry					
Paper Code: CSEN5235					
Contact Hours per week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

Course Outcomes:

Students who complete the course will demonstrate the ability to do the followings:

1. Know the common algorithms for solving well-known geometric algorithms.
2. Explain the major geometric algorithms and their analyses.
3. Apply a geometric problem or rather identify whether an algorithm for an existing geometric problem can be useful to solve the problem at hand.
4. Estimate the time and space required for implementing a geometric algorithm to solve a new problem.
5. Weigh between different geometric algorithms to solve a given problem.
6. Develop new algorithms for simple geometric problems.

Module-I:

Preliminaries: [4L]: Basic Euclidean geometry, Basic Visibility Problems , Polygons and Art Gallery Theorem, The Maximal Points Problem , The Plane Sweep Technique and applications (Segment Intersection Problem and Rectangular Union)

Convex Hull Different Paradigms [4L]: Gift wrapping, Quickhull, Graham scan, Incremental algorithm, Preparata-Hong algorithm

Module-II:

Point Location and Triangulation [4L]: Planar Point Location, Triangulation of Arbitrary Polygon, Kirkpatrick's method, trapezoidal decompositions and analysis, history DAGs

Voronoi Diagram and Delaunay Triangulation [3L]: Concepts, Delaunay triangulations. Closest Pairs, Bichromatic Closest Pairs Incremental (randomized) algorithm, Fortune's sweep, Applications.

Randomized Algorithms[3L]: Skip Lists. Randomized Incremental Construction. Planar Point Location. Persistent Data Structures.

Module-III:

Range Searching [6L]: Introduction, Orthogonal Range searching, Priority Search Trees (kd-trees, range trees and range searching, segment trees), Non - Orthogonal Range Searching, Half - Plane Range Query, Well Separated Partitioning, Adding range restrictions. Colored Range Searching

Arrangements and Duality [3L]: Point/line duality, incremental construction of arrangements and the zone-theorem, applications.

Module-IV:

Geometric Approximation [4L]: Dudley's theorem and applications, well-separated pair decompositions and geometric spanners, VC dimension, epsilon-nets and epsilon-approximations

Isothetic Geometry [3L]: Generation, Decomposition and Analysis of the Isothetic Polygon.

Matrix Searching [2L]: Concepts and its applications in different geometric optimization problems. Few applications in GIS and robot motion planning, and physical design in VLSI.

Textbooks / References:

1. Computational Geometry: Algorithms and Applications (2nd Edition), M. de Berg, M. van Kreveld, M. Overmars, O. Schwarzkopf, Springer-Verlag, 2000.
2. Computational Geometry, F. Preparata and M. Shamos, Springer-Verlag, 1985
3. Computational Geometry: An Introduction Trough Randomized Algorithms, K. Mulmuley, , Prentice-Hall, 1994
4. Discrete and Computational Geometry, S. L. Devadoss and J. O'Rourke, 2011
5. Computational Geometry Lecture Notes, David M. Mount, Department of Computer Science, University of Maryland, Fall 2002

Professional Elective IV

CSEN5241 – CSEN5245	Professional Elective IV
CSEN5241	Human and Computer Interaction
CSEN5242	Graph Algorithms
CSEN5243	Cloud Computing
CSEN5244	Algorithms for VLSI CAD
CSEN5245	Spatial Informatics and GIS

Subject Name: Human and Computer Interaction					
Paper Code: CSEN5241					
Contact Hours per week	L	T	P	Total	Credit Points
	3	0	0	3	3

Course Outcomes:

After completion of course, students would be able to:

1. Understand the structure of models and theories of human computer interaction.
2. Identify basic concepts, terminology, theories, models and methods in the field of Human Computer Interaction
3. Understand basics of interactive designing, how to prototype, iterate and refine based on the standard principles and guidelines.
4. Understand the socio organizational issues in cognitive models. Be able to identify the key players and their requirements.
5. Understand how users interact with mobile apps and widgets and design such mobile ecosystems.
6. Design an interactive web interface based on the different models studied.

Module I (7L):

Human: I/O channels – Memory – Reasoning and problem solving; The computer: Devices – Memory – processing and networks; Interaction: Models – frameworks – Ergonomics – styles – elements – interactivity- Paradigms.

Module II (11L):

Interactive Design basics – process – scenarios – navigation – screen design – Iteration and prototyping. HCI in software process – software life cycle – usability engineering – Prototyping in practice – design rationale. Design rules – principles, standards, guidelines, rules. Evaluation Techniques – Universal Design.

Module III (7L):

Cognitive models –Socio-Organizational issues and stake holder requirements –Communication and collaboration models-Hypertext, Multimedia and WWW. 8L

Module IV (11L):

Mobile Ecosystem: Platforms, Application frameworks- Types of Mobile Applications: Widgets, Applications, Games- Mobile Information Architecture, Mobile 2.0, Mobile Design: Elements of Mobile Design

Designing Web Interfaces – Drag and Drop, Direct Selection, Contextual Tools, Overlays, Inlays and Virtual Pages, Process Flow.

References:

1. Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale, “Human Computer Interaction”, 3rd Edition, Pearson Education, 2004 (UNIT I , II and III)
2. Brian Fling, “Mobile Design and Development”, First Edition ,OReilly Media Inc., 2009 (UNIT – IV)
3. Bill Scott and Theresa Neil, “Designing Web Interfaces”, First Edition, OReilly, 2009.(UNIT-V

Course Name: Graph Algorithms					
Course Code : CSEN5242					
Contact Hours per week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

Learning Objective: The main objective of the course is for students to learn some classical theorems and algorithms in this domain. It is expected that students will be able to demonstrate their knowledge of algorithms by solving concrete problems. In addition, students will be able to prove some simple facts and theorems about graphs and graph algorithms.

Course Outcomes:

Students who complete the course will demonstrate the ability to do the followings.

1. Learn the advanced concepts and key features of Graph algorithms.
2. Understand the algorithmic approach to Graph related problems.
3. Explain and analyze the major graph algorithms.
4. Employ graphs to model engineering problems, when appropriate.
5. Defend and argue the application of the specific algorithm to solve a given problem.
6. Synthesize new algorithms that employ graph computations as key components, and analyze them.
7. Hypothesize for a critical problem, where graph is involved as an absolutely necessary component.

Module I:

Connected components and transportation related graph problems

- Representation of graphs, Sub graphs, Degree Sequences, Connectivity, Cut-Vertices and Bridges, Digraphs. [1L]
- Depth First Search. DFS for undirected graphs, non-separable components and directed graphs. Topological Sorting. Strongly connected components, Tarjan's algorithm for strongly connected components. [2L]
- Eulerian tours, Characterization. De Bruijn Sequences. Eulerian Digraphs. [2L]
- Hamiltonian graphs and travelling salesman problem. Exponential-time dynamic programming for the TSP, approximation algorithms and the approximation ratio, MST-doubling heuristic, Christofides' heuristic. [4L]

Module II:

Flow networks and Bipartite graphs

- Max flow min cut theorem, max flow algorithms and their applications. [2L]
- Min cost max flow algorithm, their applications. [2L]
- Bipartite graphs, formulating bipartite maximum matching as a flow problem. [1L]

Matching and covering related graph problems

- Matchings, stable marriage problem, Gale-Shapley algorithm for stable marriage problem. [2L]
- Hopcroft–Karp algorithm. Using matchings to find vertex covers and independent sets. [2L]

Module III:

Graph Coloring, Planarity and longest path

- Graph coloring, greedy coloring, Maximal clique [2L]
- Brooks theorem, the greedy algorithm, the Welsh-Powell bound, critical graphs, chromatic polynomials, girth and chromatic number, Vizing's theorem. [2L]
- Introduction to planarity of the graph, duality of the planar graph and max cut of the planar graph. Euler's formula, Kuratowski's theorem, toroidal graphs, 2-cell embeddings, graphs on other surfaces. [4L]
- Longest path Problem, hardness and heuristic for solution. [1L]

Module IV:

Random graphs and Selected topics

- Random graphs and probabilistic methods. [2L]

- Dominating sets, the reconstruction problem, intersection graphs, interval graphs, perfect graphs, Chordal graphs.[3L]
- Maximum Clique-Minimum coloring problem in interval graph. [2L]
- Algorithms for independent set, clique and vertex coloring in Chordal graphs. [2L]

Text Books

1. Introduction to Graph Theory, Douglas B. west, Prentice Hall, 2001.
2. Graph Theory and Its Applications Jonathan L. Gross and Jay Yellen
3. Algorithm Design - Jon Kleinberg and Eva Tardos
4. Advanced graph algorithms, T.kloks

Reference Books

1. R. Diestel, "Graph Theory", Springer-Verlag, 2nd edition, 2000
2. Bela Bollobas, Modern Graph Theory, Springer, 1998

Course Name: Cloud Computing					
Course Code : CSEN5243					
Contact Hours per week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

COURSE OUTCOMES

Students who complete the course will demonstrate the ability to do the followings.

1. Identify the architecture and infrastructure of cloud computing, including SaaS, PaaS, IaaS, public cloud, private cloud, hybrid cloud.
2. Describe the core issues of cloud computing such as security, privacy, and interoperability to choose the appropriate technologies, algorithms, and approaches for the identified problems.
3. Analyze various cloud computing solutions.
4. Evaluate cloud Storage systems and Cloud security, the risks involved, its impact.
5. Apply knowledge for solving real life cloud computing problem scenario and illustrate solutions.
6. Develop appropriate cloud computing solutions and recommendations according to the applications used.

Module-1: [9L]

a. Basics of Cloud Computing [5L]:

- Defining a Cloud, Cloud Types – NIST Cloud Reference Model, Cloud Cube Model, Deployment Models (Public, Private, Hybrid and Community Clouds), Service Models – Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS)
- Characteristics of Cloud Computing – a shift in paradigm
- Benefits and Advantages of Cloud Computing

b. Concepts of Abstraction and Virtualization [4L]:

- Virtualization: Taxonomy of Virtualization Techniques
- Hypervisors: Machine Reference Model for Virtualization

Module-2: [9L]

a. Services and Applications by Type [6L]:

- IaaS – Basic Concept, Workload, Partitioning of Virtual Private Server Instances, Pods, Aggregations, Silos
- PaaS – Basic Concept, Tools and Development Environment with examples
- SaaS - Basic Concept and Characteristics, Open SaaS, examples of SaaS Platform
- Identity as a Service (IDaaS)
- Compliance as a Service (CaaS)

b. Concepts of Service Oriented Architecture (SOA) and Web Service (WS) [3L]:

- Service Oriented Architecture – Basics, Terminologies, Components, Standards and Technologies, Benefits and Challenges
- Web Services – Basics, Characteristics, Terminologies, Characteristics and Scope, Business Models

Module-3: [9L]

a. Cloud-based Storage [4L]:

- Cloud File Systems, including GFS and HDFS

b. Cloud Security [2L]:

- Cloud security concerns, security boundary, security service boundary
- Overview of security mapping
- Security of data: cloud storage access, storage location, tenancy, encryption, auditing, compliance
- Identity management (awareness of identity protocol standards)
- Risk Management and Compliance

c. Cloud Management [3L]:

- An overview of the features of network management systems and a brief introduction of related products from large cloud vendors, monitoring of an entire cloud computing deployment stack – an overview with mention of some products
- Lifecycle management of cloud services (six stages of lifecycle)
- Cloud service QoSs and maintenance

Module-4: [9L]

- **Google Web Services [2L]:** Discussion of Google Applications Portfolio – Indexed Search, Adwords, Google Analytics, Google Translate, A Brief Discussion on Google Toolkit (including introduction of Google APIs in brief), Major Features of Google App Engine Service
- **Amazon Web Services [2L]:** Amazon Web Service Components and Services: Amazon Elastic Cloud, Amazon Simple Storage System, Amazon Elastic Block Store, Amazon SimpleDB and Relational Database Service
- **Microsoft Cloud Services [2L]:** Windows Azure Platform: Microsoft's Approach, Architecture, and Main Elements, Overview of Windows Azure AppFabric, Content Delivery Network, SQL Azure, and Windows Live Services
- **Webmail Services [1L]:** Cloud Mail Services, including Google Gmail, Windows Live Hotmail, Yahoo Mail
- **Advanced topics in Cloud Computing[2L]:** Cloud Federation- Definition, popular scenario description, Replaceability and Negotiation Mechanism

Text Books:

1. Cloud Computing Bible by Barrie Sosinsky, Wiley India Pvt. Ltd, 2013
2. Mastering Cloud Computing by Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi, McGraw Hill Education (India) Private Limited, 2013
3. Cloud Computing: A Practical Approach by Anthony T. Velte, Tata Mcgraw-Hill
4. Cloud Computing by Miller, Pearson.
5. Building Applications in Cloud: Concept, Patterns and Projects by Moyer, Pearson.

References:

1. Cloud Computing (2nd Edition) by Dr. Kumar Saurabh, Wiley India
2. Cloud Computing for Dummies by Judith Hurwitz, R. Bloor, M. Kanfman, F. Halper (Wiley India Edition)
3. Enterprise Cloud Computing by Gautam Shroff, Cambridge
4. Cloud Security by Ronald Krutz and Russell Dean Vines, Wiley-India

Subject Name: Algorithms for VLSI CAD					
Paper Code: CSEN5244					
Contact Hours per week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

COURSE OUTCOMES

Students who complete the course will demonstrate the ability to do the followings.

1. Understand physical design automation, optimization techniques and data structures inside modern VLSI tools.
2. Understand how to decompose large mapping problem into pieces, including logic optimization with partitioning, placement and routing.
3. Know how to place the blocks and how to partition the blocks while for designing the layout for IC.
4. Solve the performance issues in circuit layout.
5. Analyze physical design problems and Employ appropriate automation algorithms for partitioning, floor planning, placement and routing.
6. Evaluate circuits using both analytical and CAD tools.

Module I: (10L)

Preliminaries (Data Structures and Basic Algorithms)

Data structures for Representation of Graphs, Breadth First Search, Depth First Search, Topological Sort, Spanning Tree Algorithm - Kruskal‟s and Prim‟s, Shortest path Algorithm - Dijkstra‟s Algorithm for single pair Shortest path, Floyd-Warshall algorithm for All pair Shortest path, Min cut and Max cut Algorithms

Model II: (8L)

Partitioning: Simulated Annealing, Kernighan-Lin Partitioning Algorithm, FiducciaMattheyses Algorithm. (3L)

Floor planning and Pin Assignment: Introduction, Problem Definition, Approaches to Floor planning (timing driven, Simulated Evolution, Hierarchical Tree Based, Pin assignment, Chanel assignment), Other Approaches and Recent Work. (5L)

ModuleIII: (8L)

Placement

Circuit Representation, Wire-length Estimation, Types of Placement Problem, Placement Algorithms – Constructive Placement, Iterative Improvement, Simulation Based Placement Algorithms, Partitioning Based Placement Algorithms, Other Placement Algorithms like cluster growth, Branch-and-Bound Technique

Module IV: (10L)

Global Routing

Problem Formulation, Classification of Global Routing Algorithms, Maze Routing Algorithm – Lee's Algorithm, Line-Probe algorithm, Steiner Tree based Algorithm. (5L)

Detailed Routing: Area Routing, Channel Routing – Channel Routing Model, Vertical and Horizontal Constraint Graph, Left edge Algorithm, Robust Channel Routing Algorithm (5L)

References:

1. N. Sherwani, Algorithms for VLSI Physical Design Automation, Third Edition, Kluwer, 1998
2. S. M. Sait and H. Yousuf, Iterative Computer Algorithm with Applications in Engineering, Wiley/IEEE, 2002

Subject Name: Spatial Informatics and GIS					
Paper Code: CSEN5245					
Contact Hours per week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

Learning Objective

The objective of this course is enhancing students' understanding of the physical world, knowing and communicating their relation to places in that world, and navigating through those places. Students will learn how to collect, analyze, and visualize large-scale spatial datasets while avoiding common pitfalls and building better data-intensive applications and location-aware technologies. Students will also gain a deep understanding about the fundamental research questions in individual disciplines and cross-cutting research questions requiring novel, multi-disciplinary solutions.

Course Outcomes:

Students who complete the course will demonstrate the ability to do the followings.

1. Learn the relevant Geographic Information Systems and techniques for working with geospatial data.
2. Understand how Semantic Web technology fits into the present and future evolution of GIS, and how it differs from existing data-sharing technologies, such as relational databases and the current state of the World Wide Web.
3. Explain use of Geospatial libraries to solve real-world problems with greater flexibility.
4. Employ Volunteered Geographic Information and understand how it relates to Big Geospatial Data and GIS design.
5. Recognize methods to geocode text data.
6. Synthesize and hypothesize relevant Spatial Informatics techniques to solve a variety of spatial problems.

Module 1 (9L)

Introduction and Overview of Geographic Information Systems. (4L)

Definition of a GIS, features and functions; why GIS is important; how GIS is applied; GIS as an Information System; GIS and cartography; contributing and allied disciplines; GIS data feeds; historical development of GIS.

GIS and Maps, Map Projections and Coordinate Systems (5L)

Maps and their characteristics (selection, abstraction, scale, etc.); automated cartography versus GIS; map projections; coordinate systems; precision and error.

Module 2 (9L)

Data Sources, Data Input , Data Quality and Database Concepts (5L)

Major data feeds to GIS and their characteristics: maps, GPS, images, databases, commercial data; locating and evaluating data; data formats; data quality; metadata. Database concepts and components; flat files; relational database systems; data modeling; views of the database; normalization; databases and GIS.

Spatial Analysis (4L)

Questions a GIS can answer; GIS analytical functions; vector analysis including topological overlay; raster analysis; statistics; integrated spatial analysis.

Module 3 (9L)

Making Maps (5L)

Parts of a map; map functions in GIS; map design and map elements; choosing a map type; producing a map formats, plotters and media; online and CD-ROM distribution; interactive maps and the Web.

Implementing a GIS (4L)

Planning a GIS; requirements; pilot projects; case studies; data management; personnel and skill sets; costs and benefits; selecting a GIS package; professional GIS packages; desktop GIS; embedded GIS; public domain and low-cost packages.

Module 4 (9L)

Spatial Informatics (5L)

Mathematical concepts (e.g. Euclidean space, topology of space, network space), Geo-information models (e.g. field-based, object-based), Representations (e.g. discretized, spaghetti, tessellation, voronoi diagram), Algorithms (e.g. metric and Euclidean, topological, set-based, riangulation, graph-based), Data Structures and access methods (e.g. space filling curves, quad-trees, R-tree), Analysis (e.g. spatial query languages, spatial statistics, spatial data mining).

Location based services (4L)

Overview, Positioning Technologies, Mapping, Applications. Spatial Networks: Representation, Access Methods.

Text Book:

1. Concepts and Techniques of Geographic Information Systems by C.P. Lo and Albert K.W. Yeung, Prentice Hall, 2006.
2. Spatial Databases: A Tour by Shashi Shekhar and Sanjay Chawla, Prentice Hall, 2003.

References:

1. Kuhn, Werner. “Geospatial semantics: why, of what, and how?” Journal on Data Semantics III (pp. 1-24). Berlin, Germany, Springer-Verlag Lecture Notes in Computer Science Vol. 3534 (2005).
2. Fonseca, Frederico. "Geospatial semantic web." Encyclopedia of GIS. Springer US (2008), 388-391
3. Goodchild, Michael F., and Linna Li. "Assuring the quality of volunteered geographic information." Spatial Statistics 1 (2012): 110-120

Subject Name: Advanced Algorithms Lab					
Paper Code: CSEN5251					
Contact Hours per week	**L**	**T**	**P**	**Total**	**Credit Points**
	0	**0**	**4**	**4**	**2**

Course Outcomes/Learning Objectives:

- On completion this course, students are expected to be capable of understanding basic ability to analyze algorithms and to determine algorithm correctness and time efficiency class.
- Beside this students should be able to understand basic features of different algorithm design paradigms like divide and conquer, greedy, dynamic programming etc.
- Last but not the least, students will be able to apply and implement learned algorithm design techniques and data structures to solve various real life problems.

In this laboratory Students should run all the programs using C programming language on LINUX platform and then estimate the running time of their programs in best, worst and average case situations for large dataset.

A tentative outline of the laboratory is given below:

- Divide and Conquer: Find Maximum and Minimum element from a array of integer using Divide and Conquer approach
- Divide and Conquer: Implement Quick Sort using Divide and Conquer approach. Check the running time for different positions of pivot elements. Implement the randomized version of quick sort
- Dynamic Programming: Find the minimum number of scalar multiplication needed for chain of Matrices
- Implement Single Source shortest Path for a graph (Dijkstra and Bellman Ford Algorithm)
- Dynamic Programming: Implement all pair Shortest path for a graph (Floyd- Warshall Algorithm)
- Greedy method: implement fractional Knapsack Problem, MST by Prim's algorithm
- Greedy method: Implement MST by Kruskal's algorithm by using Union operation on Disjoint data Structures.
- Graph Traversal Algorithm: Implement Depth First Search (DFS), application of DFS (do topological sorting, identify strongly connected components)
- Implement KMP algorithm for string matching
- Implement Ford-Fulkerson algorithm to get maximum flow of a given flow network.

Subject Name: Soft Computing Lab					
Paper Code: CSEN5252					
Contact Hours per week	**L**	**T**	**P**	**Total**	**Credit Points**
	0	**0**	**4**	**4**	**2**

COURSE OUTCOMES

- An understanding of fundamental concepts and methods of machine learning and its applications.
- An ability to analyze and evaluate simple algorithms for pattern classification.
- An ability to design simple algorithms for pattern classification, code them with Python programming language and test them with benchmark data sets.

A tentative outline of the laboratory is given below:

1. Introduction to Matlab/Python, Arrays and array operations, Functions and Files. Familiarization with a few ML Tools such as Excel, WEKA, R, Python and TensorFlow
2. Study of neural network toolbox and fuzzy logic toolbox.
3. Simple implementation of Artificial Neural Network and Fuzzy Logic
4. Implementation of latest soft computing techniques using one of the above tools.
5. Regression (single and Multiple Variables) linear and non-liner;
6. Logistic regression
7. Classifiers: K-NN, Naïve Bayes Classifier, Perceptron, Multi Layer Perceptron
8. Clustering Algorithms: K-Means , DB-Scan
9. Applications of ANN and SVM using ML tools

M. Tech. Detailed Syllabus - Semester III

Professional Elective V

CSEN6131 – CSEN6139	Professional Elective V
CSEN6131	Mobile Applications and Services
CSEN6132	Compiler for HPC
CSEN6133	Computational Complexity
CSEN6134	Fault Tolerant Computing
CSEN6135	Approximation Algorithms
CSEN6136	Randomized Algorithms
CSEN6137	Information Retrieval
CSEN6138	Social Network Analysis
CSEN6139	Quantum Computing

Subject Name: Mobile Applications and Services					
Paper Code: CSEN 6131					
Contact Hours per week	L	T	P	Total	Credit Points
	3	0	0	3	3

Course Outcomes:

1. Understand the methodology and syntax of implementing applications for mobile devices working in the Android and iOS Platform
2. Understand the concept of RESTful and Non-RESTful apps
3. Create and Incorporate Web/Cloud Services
4. Understand the working of Mobile Sensors and develop apps to interact with them
5. Learn Security and Trust Management
6. Develop the understanding of Privacy and Ethics

Module I: Introduction to the Mobile Device Architecture and Android Architecture (9L)
Wireless Application Protocol (WAP): The Mobile Internet standard, WAP Gateway and Protocols. The Three-Tiered Architecture: Model-View-Controller, Client / Server Architecture. Mobile Architectures: Thick Client, Thin Client. Android Architecture: Introduction to the Android Architecture, Android Studio, Building Activities, Intents and Services, Lifecycle Management.

Module II: Introduction to MAC and iOS Architecture (9L)
A History of iOS and Mac Development. Xcode and iOS Architecture. iOS App Components.

Module III: Data Management and Sensors (9L)
Android Data Management. iOS Data Management. REST: Service Oriented Arch, Web Services. Security / OAUTH. Sensor Ethics, Hardware and Sensors.

Module IV: User Interface (9L)
User Interface Design: UI Design, Accessibility, Android UI and Material Design Principles, Apple UI and HIG. Wireframing. Privacy Policies and Analytics, Monetization. Cross-Platform Development. Wearables, AR, VR, and Mobile.

References:

1. https://www.infoworld.com/article/2623833/mobile-development-101--what-you-need-to-know.html
2. https://developer.android.com/guide/index.html
3. https://developer.apple.com/library/archive/referencelibrary/GettingStarted/DevelopiOSAppsSwift/
4. https://developer.apple.com/documentation/
5. https://developer.android.com/guide/topics/data/data-storage.html
6. https://developer.apple.com/library/archive/documentation/Cocoa/Conceptual/CoreData/nsfetchedresultscontroller.html
7. https://www.restapitutorial.com/
8. http://tutorials.jenkov.com/oauth2/overview.html
9. https://developer.android.com/guide/topics/sensors/sensors_overview.html
10. http://w3c.github.io/Mobile-A11y-TF-Note/#
11. https://material.io/design/introduction/#
12. https://developer.apple.com/design/human-interface-guidelines/ios/overview/themes/
13. https://developer.android.com/training/design-navigation/wireframing.html
14. https://www.docracy.com/mobileprivacy/
15. https://toucharcade.com/2015/11/13/paid-games-dont-work-for-developers-heres-why-the-carter-crater/
16. https://cordova.apache.org/docs/en/latest/
17. https://gizmodo.com/all-of-the-best-augmented-reality-tricks-your-iphone-ca-1805804012

Subject Name: Compiler for HPC					
Paper Code: CSEN6132					
Contact Hours per week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

Course Outcome:

After completion of the course, students would be able to:

1. Remember the basic concepts of code generation and machine independent optimizations.
2. Learn various scheduling techniques and register allocation for exploiting Instruction Level Parallelism.
3. Be familiar with some well-known memory locality optimizations and demonstrate their understanding of locality optimization in different algorithms.
4. Apply the concept and compiler techniques for exploiting Data Level Parallelism.
5. Estimate the scope and level of parallelism in any application.
6. Design new basic block scheduling algorithms for data dependence graph applying the concept gained from the course.

Module-I (8L)

Review of code generation: optimization of basic blocks, peephole optimization. Registers allocation and assignment: Global register allocation, Register allocation by graph colouring. Optimal code generation for expressions: Ersov numbers, Evaluation expressions with insufficient supply of registers.

Module-II (10L)

Instruction level parallelism: Processor architectures; Code scheduling constraints: data dependence, Control dependence; Basic block scheduling: Data dependence graph, List scheduling of basic blocks, Prioritized Topological orders; Global code scheduling: Primitive, upward, and downward code motion; Introduction to Global scheduling algorithms; Software pipelining: Software pipelining of loops, Scheduling acyclic and cyclic data dependence graphs.

Module-III (8L)

Memory hierarchy of a computer and its optimization: reducing fragmentation. Basic introduction to garbage collection: reachability, Reference counting garbage collectors; Introduction to trace-based collector: a basic Mark-and-Sweep collector, Optimizing Mark-and-Sweep; Mark-and-Compact garbage collector; Parallel and concurrent garbage collection; Partial object relocation.

Module-IV (10L)

Optimizing for parallelism and locality: Multiprocessors and parallelism in applications, Loop-level parallelism, Data locality. Optimization issues in Matrix multiplication algorithm, Different types of Data reuses; Identification of Synchronization-free parallelism; Synchronization between parallel loops; Pipelining: basic introduction, Parallelism with minimum synchronization; Locality optimization: Temporal locality of computed data, Partition interleaving.

References

1. Compilers: principles, techniques, and tools - Alfred V Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, Pearson Education.
2. Advanced compiler design implementation - Steven S. Muchnick.
3. Optimizing compilers for modern architectures: a dependence-based approach – Randy Allen, Ken Kennedy.

Subject Name: Computational Complexity					
Paper Code: CSEN6133					
Contact Hours per week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

Course Outcomes:
Students who complete the course will demonstrate the ability to do the followings.

1. Learn and understand the fundamentals of various notions in computational complexity theory to classify computational problems.
2. Understand the important complexity classes, relationship among themselves, and some typical problems in the field.
3. Understand and explain the techniques used in analysis of computational complexity.
4. Classify decision problems into appropriate complexity classes such as P, NP, PSPACE and complexity classes based on randomized machine models.
5. Gain the concept to classify optimization problems into appropriate approximation complexity classes.
6. Apply complexity theory and the concept of interactive proofs in the analysis of optimization problems in different domains.

Module-I (10L)
Computational Models: Problems, Computability, Algorithms, and Complexity; Introduction to P and NP; Review of Turing machines and universal Turing machines; Turing machines Logic (Boolean logic, circuits).

Module-II (10L)
P, NP, coNP, and NP-Completeness; P vs. NP, NP vs. coNP; NP-completeness of SAT and other problems; Search vs. decision and self-reducibility, Complexity classes (hierarchy theorem, P, NP, Co-NP); Reduction and completeness.

Module-III (8L)
Interactive proof systems; Polynomial hierarchy; Diagonalization: Time/space hierarchy theorems; Ladner's theorem; Space complexity: PSPACE and PSPACE-completeness; NL and NL-completeness.

Module-IV (8L)
Randomized computation: Basic concept, Definitions and relation among the randomized classes RP, coRP, PP, BPP; Relation of BPP to the polynomial hierarchy and non-uniform computation; Nondeterministic Space Classes: Logarithmic space; Polynomial space, Savitch's Theorem; Exponential time and space. A PSPACE complete problem- quantified Boolean formula problem (QBF).

References

1. Michael Garey and David S. Johnson: Computers and Intractability: A Guide to the Theory of NP-Completeness. New York: W. H. Freeman and Co., 1979.
2. Sanjeev Arora and Boaz Barak: Computational Complexity: A Modern Approach. Cambridge University Press, 2009.
3. Christos H. Papadimitriou: Computational Complexity, Addison-Wesley Longman.
4. Michael Sipser: Introduction to the Theory of Computation, PWS Publishing.
5. John E. Hopcroft and Jeffrey D. Ullman, Introduction to Automata, Languages and Computation, Addison-Wesley, 1979.
6. J. Balcazar, J. Diaz, and J. Gabarro, Structural Complexity, Volumes I and II, Springer.

Subject Name: Fault Tolerant Computing					
Paper Code: CSEN6134					
Contact Hours per week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

Course Outcomes:

Students who complete the course will demonstrate the ability to do the followings:
1. understand reliability and fault tolerance in electronic system.
2. understand different types of defects, faults, errors and hazards.
3. know how to create a stochastic modeling of failure / hazard.
4. solve the faults / hazards.
5. analyze reliability modeling of redundancy systems.
6. evaluate reliability, availability, serviceability or real time systems.

Module I: Preliminaries (Data Structures and Basic Algorithms) (9L)
Principles of Fault Tolerance; Reliability Requirements; Hardware F-T Techniques; System Abstractions; Software Structuring Schemes;

Module II: (9L)
Techniques for Different Stages of Fault Tolerance; Techniques for Different Types of Faults;

Module III: (9L)
Fault Tolerance in Distributed Systems; Fundamental Problems in Coordination; Communication and Remote Operation over Unreliable Channels; Fault Tolerant Control and Coordination Algorithms Design;

Module IV: (9L)
F-T System Abstractions/Functions; System Mechanisms for F-T; Fault Tolerant Programming Paradigms; Modeling and Analyzing F-T Distributed Systems.

References:

1. "Reliability of computer systems and networks" by Martin L. Shooman, John Wiley and Sons Inc., 2002, ISBN 0-471-29342-3.
2. Anderson, T., and P.A. Lee, Fault-Tolerant Principles and Practices, Prentice-Hall Int'l., London, 1981.
3. Hwang, K., and F.A. Briggs, Computer Architecture and Parallel Processing, McGraw-Hill, 1984.
4. Jalote, P. Fault-Tolerance in Distributed Systems, ISBN 0-13-301367-7, Prentice-Hall, 1994.
5. Johnson, B.W., Design and Analysis of Fault-Tolerant Systems, Addison Wesely, 1989.
6. Leveson, Nancy G., Safeware, system safety and computers, Addison Wesely, 1995.
7. Pradhan, D.K., Fault-Tolerant Computing -- Theory and Techniques, (2 Volumes), Prentice-Hall, 1986.
8. Pradhan, Dhiraj K., Fault-Tolerant Computer System Design, ISBN 0-13-057887-8, Prentice-Hall PTR , 1996.
9. Sahner, R.A., K.S. Trivedi and A. Puliafito, Performance and Reliability Analysis of Computer Systems, Kluwer Academic Publishers, 1996.
10. Sieworek, and R.S. Schwarz, The Theory and Practice of Reliable System Design, Digital Press, 1982.
11. Storey, Neil, Safety-Critical Computer Systems, Addison Wesely, 1995.
12. Martin L. Shooman, Reliability of computer systems and networks, John Wiley and Sons Inc., 2002, ISBN 0-471-29342-3. Trivedi, K.S., Probability and Statistics with Reliability, Queuing, and Computer Science Applications, Prentice-Hall, 1982.

Subject Name: Approximation Algorithms					
Paper Code: CSEN6135					
Contact Hours per week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

Course Outcomes:

On completion of this course, students would be able to:

1. Remember the approach of designing different approximation algorithms to solve various hard problems.
2. Analyze a given real life problem to determine it hardness, and then define an approximation algorithm for it.
3. Learn the limits of approximation and the basic ways of proving hardness of approximation
4. Choose appropriate approximation algorithms and use it for a specific hard problem.
5. Hypothesize for a critical problem, where graph is involved as an absolutely necessary component.
6. Gather some knowledge about the recent developments in the area of approximation algorithmic design

Module 1 (9L)

NP-Completeness: Polynomial time, NP-Hardness, NP-Completeness and reducibility, NP-Completeness proofs. Approximation Algorithms: Fundamentals and Concepts. Performance Ratio. Polynomial approximation scheme (PTAS), Fully polynomial time approximation scheme (FPTAS).

Module 2 (9L)

Approximation algorithms for scheduling. List scheduling. Job scheduling with deadlines. Identical aparallel machines. Unrelated parallel machines. Bin Packing. Next fit, first fit, online and offline algorithms. Average case analysis.

Module 3 (9L)

Approximate covering and poacking, set cover, vertex cover, independent set.
Approximation algorithms for highly connected subgraphs. Weighted and unweighted vertex connectivity. Weighted and unweighted edge connectivity. Strong connectivity.

Module 4 (9L)

Approximation Algorithms for Geometric problems. Euclidean TSP, Steiner tree problems, Steiner ratio, Minimum weight triangulation with steiner points, Clustering, K-minimum spanning tree, polygon separation, point set separation.
Hardness of approximations. Inapproximability results. PCP theorem. PCP and inapproximability of MAX-3SAT.

Text Books:

1. Approximation Algorithms by Vijay Vazirani, (Springer, 2001)
2. Approximation Algorithms for NP-Hard Problems by Dorit S. Hochbaum (PWS Publishing Company, 1997)

Subject Name: Randomized Algorithms					
Paper Code: CSEN6136					
Contact Hours per week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

Course Outcomes:

After completion of the course, students would be able to:

1. Remember the basic concepts of probability calculus in algorithmic context and apply the probabilistic method to show the existence of certain combinatorial objects
2. Demonstrate their understanding of algorithmic randomization for a given problem.
3. Understand and use suitable mathematical tools to design randomized algorithms and analyze their performance.
4. Calculate proper upper bounds for the expected running time of simple randomized algorithms.
5. Estimate how a randomized algorithm performs asymptotically better than the best known exact deterministic algorithms for that problem.
6. Design simple randomized algorithms that run fast or that return the correct output with high probability.

Module 1 (9L)
Introduction. Basic Probability Theory. Moments and deviations, Markov and Chebyshev inequalities. Tail Estimates and the Chernoff Bound. Conditional Expectation and Martingales.
The Probabilistic Method. Markov Chains and Random Walks.

Module 2 (9L)
Sorting: Randomized Quicksort. Analysis. Comparison with average case analysis of deterministic Quicksort.
Searching: Skip Lists.

Module 3 (9L)
Randomized Incremental Construction. Randomized Data Structures for dynamic data.
Randomized Graph Algorithms.

Module 4 (9L)
Implementation issues. De-randomization. Applications: Algorithms for Data Streams.

Text Book:

1. Randomized Algorithms by Rajeev Motwani and Prabhakar Raghavan. (Cambridge University Press).

References:

1. Computational Geometry: An Introduction through Randomized Algorithms by Ketan Mulmuley, Prentice Hall.

Course Name: Information Retrieval					
Course Code: CSEN6137					
Contact hrs per week:	L	T	P	Total	Credit points
	3	0	0	3	3

Course Objectives:
The objective of the course is to introduce information retrieval models and query languages. Application of web search and information retrieval in social networks is also included.

Course Outcomes:
After completion of course, students would be able to:
1. Identify basic theories and analysis tools as they apply to information retrieval.
2. Develop understanding of problems and potentials of current IR systems.
3. Learn and appreciate different retrieval algorithms and systems.
4. Apply various indexing, matching, organizing, and evaluating methods to IR problem
5. Be aware of current experimental and theoretical IR research.
6. Analyze and design solutions for some practical problems.

Module I: (9L)
Information retrieval model, Information retrieval evaluation; Document Representation – Boolean Model, Posting Lists, Inverted Indices, Skip Lists; Query languages and query operation – proximity search, Phrase Queries Meta-data search; Tolerant Retrieval – B-Trees, Permuterm Index, Edit Distance – Different variations

Module II: (9L)
Indexing Construction and Searching – BSBI, SPIMI, Heap's Law Zipf's Law; Scoring and ranking feature vectors, tf-idf various schemes; Evaluation and computations of scores and ranked retrieval; Relevance feedback

Module III: (9L)
Text and multimedia languages, Language Models – Query Likelihood Models; Text Classification and Naïve Bayes – Bernoulli model, feature selection; Vector Space Classification – kNN, Rocchio Classification

Module IV: (9L)
Flat Clustering – K means, K medoids, Evaluation of clustering, Models for clustering; Hierarchical Clustering – Single Link, Complete Link, Group Average and Centroid, Inversion Points, DivisiveClustering – Basics; Latent Sematic Analysis – SVD, Low Rank Approximations;
Web Search Basics, Link Analysis – Page Rank, HITS.

References:

1. C. D. Manning, P. Raghavan and H. Schütze, Introduction to Information Retrieval, Cambridge University Press, 2008 (available at http://nlp.stanford.edu/IR-book).
2. Chakrabarti, S. (2002). Mining the web: Mining the Web: Discovering knowledge from hypertext data. Morgan-kaufman.
3. B. Croft, D. Metzler, T. Strohman, Search Engines: Information Retrieval in Practice, AddisonWesley, 2009 (available at http://ciir.cs.umass.edu/irbook/).
4. R. Baeza-Yates, B. Ribeiro-Neto, Modern Information Retrieval, Addison-Wesley, 2011 (2nd Edition).

Subject Name: Social Network Analysis					
Paper Code: CSEN6138					
Contact Hours per week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

Course Outcomes:

1. Students should be able to demonstrate basic knowledge of social networks and related application-oriented models.
2. Students should be able to understand applications of graph algorithms in social networks.
3. Students should be able to write programs to implement the related social network analysis algorithms when necessary.
4. Students should be accustomed to various network related libraries (in Python/Java/R/C++) to implement social network theories.
5. Students will get an exposure to the present state-of-the-art algorithms and methods in the area of social networks.
6. Exposure to the state-of-the-art algorithms should help the students in pursuing research in areas related to social networks.

Module I. Introduction [9L]

Motivating challenges in analysing social networks. (1L)

Measures and Metrics (4L): Degree centrality, Eigenvector centrality, Katz centrality, PageRank, hubs and authorities (HITS), closeness centrality, betweenness centrality, groups of vertices, transitivity, reciprocity, signed edges and structural balance, similarity, homophily and assortative mixing

Large Scale Structure of Networks (4L):

Components, shortest paths and the small world effect, degree distributions, power laws and scale-free networks, distributions of centrality measures, clustering coefficients

Module II. Random Networks [9L]

Understanding mean number of edges, mean degree, degree distribution, clustering coefficient, giant component, small components, and average path lengths for the following models-

Erdos-Renyi Network (3L); Small-world networks and Watts-Strogatz model (3L); Preferential attachment and Barabasi-Albert model (3L)

Module III. Propagation of Information in Netowrks [6L]

Contagion Models (3L): Models of disease spread – SI, SIS, SIR, SIRS and related literature. Outbreak detection.

Influence Maximization (3L): Influence spread models - independent cascade model, linear threshold model. Maximizing propagation of influence under different setups – greedy approximation algorithm by Kempe et. al. and related literature.

Module IV. Community Detection [12L]

What is a community? Notion of disjoint and overlapping communities. Goodness measures – modularity. Benchmarks and comparing with the benchmarks (F-measure, NMI, Omega index) (2L)

Strength of weak ties and related models. (1L); Clique Percolation model (1L); Modularity maximization, Clauset-Newman-Moore (CNM) method, Louvain Method (3L); Label propagation algorithm and its variants (2L); Random walks, Entropy-based method: Infomap (2L); Community preserving sparsification of social networks (1L)

Text Books :

1. Networks: An Introduction by Mark Newman. Oxford University Press.

Reference Books :

1. Networks, Crowds and Markets: Reasoning About a Highly Connected World by David Easley and Jon Kleinberg.

Subject Name: Quantum Computing					
Paper Code: CSEN6139					
Contact Hours per week	**L**	**T**	**P**	**Total**	**Credit Points**
	3	**0**	**0**	**3**	**3**

Course Outcomes:

Students who complete the course will demonstrate the ability to do the followings.

1. Understand the major mathematical representations of quantum operations,
2. Distinguish between classical and quantum computation
3. Describe a few key applications of quantum computing
4. Implement basic quantum algorithms,
5. Explain quantum decoherence in systems for computation,
6. Understand and describe quantum information concepts
7. Identify key aspects of quantum supremacy over conventional computation.

Module I: (9 L)

Introduction and Overview : Brief history and postulates of quantum theory; Recapitulation of the basic principles of classical computation; Dirac Notation, Probability amplitudes.

Quantum Mechanics: Superposition and wave function collapse; Qubits; Quantum measurements – Positive operator valued measures and Projective measurements; Heisenberg's Uncertainty principle; Brief discussion on the difference between classical and quantum probability.

Module II: (9L)

Linear Algebra and Hilbert Spaces: Basis vectors- Orthogonal and Orthonormal vectors; Inner product spaces, Completeness and Separable Hilbert spaces; Unitary operations and Projectors; Tensor Products.

Fundamental quantum notions: No-cloning theorem; Quantum entanglement; Quantum nonlocality – Bell's inequality.

Module III: (11L)

Quantum Circuits: Pauli and Hadamard gates; Phase, CNOT, Toffoli gates; Universal Quantum gates; Prototype examples - Quantum Teleportation and super dense coding; Reversible computing.

Quantum Algorithms: Deutsch-Josza algorithm; Simon's problem; Quantum Fourier transform; Shor's period-finding algorithm; Grover's algorithm for searching; Basic quantum cryptography - BB84 and E91 protocol.

Module IV: (7L)

Quantum Computers: Physical qubits; Noise and Decoherence

Basic aspects of quantum information theory: Shannon and von-Neumann entropy; Conditional entropy , relative entropy and Mutual information

References:

1. N. David Mermin, Quantum Computer Science – An Introduction, Cambridge University Press, 2007.
2. Michael A. Nielsen and Issac L. Chuang, "Quantum Computation and Information", Cambridge (2002).
3. Riley Tipton Perry, "Quantum Computing from the Ground Up", World Scientific Publishing Ltd (2012).
4. Scott Aaronson, "Quantum Computing since Democritus", Cambridge (2013).
5. P. Kok, B. Lovett, "Introduction to Optical Quantum Information Processing", Cambridge (2010).

Lecture Notes:

1. John Preskill's lecture notes: http://www.theory.caltech.edu/people/preskill/ph229/
2. David Mermin's lecture notes: http://people.ccmr.cornell.edu/_mermin/qcomp/CS483.html

Open Elective

CSEN6121	**Business Analytics**
ECEN6122	**Design of Embedded Systems**
INFO6123	**Information Theory and Coding**
ECEN6124	**Automation in VLSI Design**
MATH6121	**Optimization Techniques**
AEIE6122	**Intelligent Control**

Please refer to the "M Tech 3rd Sem Open Electives" document for detail syllabus.