Courses Offered by

Department of Computer Science and Engineering

Course - List

200 - Level Courses

• Compulsory Courses (CSE and AI&DS)

Course Title	**Data Structures and Algorithms**	Course No.	**CSxxx**
Department	Computer Science and Engineering	Structure (L-T-P [C])	3-0-2 [4]
Offered for	B.Tech of all branches	Type	Compulsory
Prerequisite	Introduction to Computer Programming	Antirequisite	None

Objectives

1. To introduce and practice the implementation of various data structures used for indexing, searching, and sorting operations.
2. To introduce basic mathematical techniques for algorithm analysis and design.

Learning Outcomes

1. Ability to design and implement appropriate data structures for indexing, searching, and sorting operations for real-world problems.
2. Designing of new algorithms using standard data structures.
3. Analyzing the time and space complexities of standard data structures and basic algorithms.

Contents

Algorithm analysis and complexity: Big/little -Oh, Omega, Theta notation, Recurrence equations. (2 Lectures)
Abstract data types: Linear data structures, Tree, Binary trees, Tree traversal, Applications. (7 Lectures)
Search trees: Binary search trees, Balanced search trees, AVL trees, B-Trees. (5 Lectures)
Heaps: Binary Heap, Heap order property and min/max heaps. (3 Lectures)
Sets: Disjoint set ADT, Basic operations on Sets, Union/Find algorithm. (2 Lectures)
Sorting algorithms: Bubble sort, Selection sort, Bucket sort, Insertion sort, Overview of Divide-and-conquer, Quick sort, Merge sort. (6 Lectures)
Hashing: Hash tables and operations, Hash function, Open and closed hashing, External and internal hashing, Collision resolving methods, Rehashing. (5 Lectures)
Graph algorithms: Definitions, Branch and bound, Backtracking, Representation, Traversal, Shortest-path algorithms, Minimum Spanning Tree algorithm, Topological sorting. (8 Lectures)
Greedy techniques and Dynamic programming (4 Lectures)

Laboratory

1. Implementation of data structures using C programming language.
2. Practically verifying and comparing run-time performance and asymptotic behavior of various data structures and related algorithms.
3. Applications of data structures from real-life scenarios.

Text Book

M. A. WEISS (2002), Data Structures and Algorithm Analysis in C, Addison-Wesley, 2nd Edition.

Reference Book

T. H. CORMEN, C. E. LEISERSON, R.L. RIVEST, C. STEIN (2009),Introduction to Algorithms, MIT Press, 3rd Edition.

Course Title	**Design and Analysis of Algorithms**	Course No.	**CSLxxx**
Department	Computer Science and Engineering	Structure (L-T-P [C])	3-1-0 [4]
Offered for	B.Tech CSE, AI&DS	Type	Compulsory
Prerequisite	Data Structures and Algorithms	Antirequisite	None

Objectives
1. To introduce and implement various techniques for designing algorithms and advanced data structures.
2. To learn space and time complexity analysis of algorithms.

Learning Outcomes
1. Ability to choose and implement appropriate algorithm design techniques for solving problems.
2. Understand how the choice of data structures and algorithm design methods impact the performance of programs.
3. Ability to analyze the worst-case and average-case behaviour of algorithms in terms of time and memory requirements.

Contents
Reasoning About Algorithms: P, NP, NP-completeness, Reductions, Complexity analysis. (5 lectures)
Graph Algorithms: Strongly-connected components, Kosaraju's algorithm 1 and 2, Applications. (4 lectures) ***Greedy Techniques:*** Local versus Global optimality, Interval scheduling, Exchange arguments. (5 lectures)
Divide-and-Conquer: Optimality, Recursive algorithms, Divide-and-Conquer recurrences, The Master Theorem and applications, Non-uniform recurrences. (6 lectures)
Dynamic Programming: Reusing sub-computations (Sequence alignment, Bellman-Ford algorithm), Precomputing (Floyd-Warshall algorithm, Johnson's algorithm), Combinatorial problems. (Knapsack) (6 lectures)
Linear Programming: Canonical and standard forms, Feasibility and optimization, Simplex algorithm. (5 lectures)
Approximation Algorithms: Relative approximations, PAS and FPAS scheduling. (4 lectures)
Randomized Algorithms: Random guess (Quick select), Random guess with high confidence (Karger's min-cut algorithm), Storing associative data (Hashing), Error bounds. (7 lectures)

Text Book
T. H. CORMEN, C. E. LEISERSON, R.L. RIVEST, C. STEIN (2009), Introduction to Algorithms, MIT Press, 3rd Edition.

Reference Book
J. KLEINBERG, E. TARDOS (2005), Algorithm Design, Pearson Education, 1st Edition.

Course Title	**Human-Machine Interaction**	Course No.	**CS2xxx**
Department	Computer Science and Engineering	Structure (L-T-P [C])	0-0-4 [2]
Offered for	B.Tech (CSE, AI&DS)	Type	Compulsory
Prerequisite	None	Antirequisite	None

Objectives
To provide a theoretical and practical understanding of human-machine interaction (HMI) design, including concepts of user centered and design thinking, usability, interfaces, rapid prototyping, and evaluation.

Learning Outcomes
1. The students will have:
2. A broad understanding of human-machine interaction and latest technologies.
3. Understanding of the perceptual and cognitive basis of human-machine interaction.
4. Knowledge of user centered design and techniques for rapid prototyping.
5. Knowledge of assessing usefulness and usability of a design
6. Introduction to approaches for gathering and analyzing interaction data, and conveying design concepts.

Lectures (discussions in the lab and web-based flip pedagogy)
1. Theories and Research Methods in HMI
2. Understanding User Experience
3. Foundations and Concepts of Interaction
4. Prototyping Techniques
5. Prototyping Evaluation
6. Communicating User Needs and Requirements
7. Speculative Design
8. Case Studies

Laboratory and Assignments (primary approach)
1. Find a poorly designed item (anything).
   Submit either a picture or sketch and describe why it is poorly designed. (Week 1)
2. Heuristic Evaluation and Interview (Week 1)
3. User Scenarios, Personas and Storyboards ((Week 2)
4. User Journeys (Week 3)
5. Wireframes: Paper and Digital Prototyping (Week 4)
6. Prototype Evaluation Study Design (Week 5, 6)
7. Value Sensitive Design Evaluation (Week 7, 8)
8. Design visual Interfaces (laptop, mobile) - e.g. gesture based (Week 9, 10)
9. Design voice interfaces - e.g. speech chatbot (Week 11, 12)
10. Design multimodal interactions (Week 13, 14)

Text Book
H. SHARP, J. PREECE, Y. ROGERS (2019), Interaction Design: Beyond Human-Computer Interaction, Wiley, 5th Edition.

Course Title	**Maths for Computing**	Course No.	**CSLXXX**
Department	Computer Science and Engineering	Structure (L-T-P [C])	3-1-0 [4]
Offered for	B.Tech CSE, AI&DS	Type	Core
Prerequisite	None	Antirequisite	None

Objectives
1. To learn about languages, grammars, and computation models
2. To learn about computability
3. To learn basics of parsing techniques

Learning Outcomes:
The students are expected to have the ability to:
1. Model computer science problems using discrete mathematical structures
2. Distinguish between computable and non-computable problems
3. Develop understanding of properties of languages and design parsers

Contents
Discrete Structures: Can computers solve every problem? The limits of computing, Set theory, Relations and functions, Propositional logic, First-order logic, Counting techniques. (12 Lectures)
Proof techniques: Constructive, Contraposition, Contradiction, Mathematical Induction (5 Lectures)
Graph Theory: Properties of graphs, Graph matching and coloring. (6 Lectures)
Automata Theory: DFAs, NFAs, Equivalence of DFAs and NFAs, Closure properties of regular languages, Regular expressions, Equivalence of regular expressions and NFAs, Nonregular languages, Context-Free Grammars, Context-Free Languages. (10 Lectures)
Turing Machine: Introduction, Designing Turing machines, The Universal Turing machine. (5 Lectures)
Parsing Techniques: *LR, LALR, Shift-Reduce parsers. (4 Lectures)*

Text Books
M. SIPSER (2014), Introduction to the Theory of Computation, Cengage Learning, 3rd Edition.
K.H. ROSEN (2018), Discrete Mathematics and its Applications, McGraw-Hill, 2018, 8th Edition.

References
J.E. HOPCROFT, R. MOTWANI, J.D. ULLMAN (2008), Pearson, Introduction to Automata Theory, Languages, and Computation, 3rd Edition.
R. JOHNSONBAUGH (2017), Discrete Mathematics, Prentice Hall, 8th Edition.

Self-learning material
Stanford CS103: http://web.stanford.edu/class/cs103/

Course Title	**Pattern Recognition and Machine Learning**	Course No.	**CSL2xx**
Department	Computer Science and Engineering, AI&DS	Structure (L-T-P [C])	3-0-2
Offered for	B.Tech (CSE, AI&DS, EE)	Type	Compulsory
Prerequisite	Introduction to Computer Science, Probability, Statistics and Stochastic Processes	Antirequisite	Introduction to Machine Learning

Objectives
1. To understand various key paradigms for pattern classification and machine learning approaches
2. To familiarize with the mathematical and statistical techniques used in pattern recognition and machine learning.
3. To understand and differentiate among various pattern recognition and machine learning techniques.

Learning Outcomes

The students are expected to have the ability to:
1. To formulate a machine learning problem
2. Select an appropriate pattern analysis tool for analyzing data in a given feature space.
3. Apply pattern recognition and machine learning techniques such as classification and feature selection to practical applications and detect patterns in the data.

Contents

Introduction: Definitions, Datasets for Pattern Recognition, Different Paradigms of Pattern Recognition and
Machine Learning, Data Normalization, Hypothesis Evaluation, VC-Dimensions and Distribution, Bias-Variance Tradeoff, Regression (Linear) (8 Lectures)
Discriminative Methods: Distance-based methods, Linear Discriminant Functions, Decision Tree, Random Decision Forest and Boosting (5 Lectures)
Bayes Decision Theory: Bayes decision rule, Minimum error rate classification, Normal density and discriminant functions, Bayesian networks (7 Lectures)
Parameter Estimation: Maximum Likelihood and Bayesian Parameter Estimation (3 Lectures)
Feature Selection and Dimensionality Reduction: PCA, LDA, ICA, SFFS, SBFS (4 Lectures)
Artificial Neural Networks: MLP, Backprop, and RBF-Net (4 Lectures)
Kernel Machines: Kernel Tricks, Support Vector Machines (primal and dual forms), K-SVR, K-PCA (6 Lectures)
Clustering: k-means clustering, Gaussian Mixture Modeling, EM-algorithm (5 Lectures)

Laboratory

Programming labs on Normalization, Visualization, Evaluation, Regressions, Decision Tree,
Bayes, Parameter Estimation, PCA, LDA, ANN, SVM, K-means, GMM, Feature Selection

Text Book
1. R. O. DUDA, P. E. HART, D. G. STORK (2000), Pattern Classification, Wiley-Blackwell, 2nd Edition.

Reference Books
1. C. M. BISHOP (2006), Pattern Recognition and Machine Learning, Springer-Verlag New York, 1st Edition.
2. T. M. MITCHELL (2017), Machine Learning, McGraw Hill Education, 1st Edition.

Self-learning Material
1. Introduction to Machine Learning, NPTEL Course Material, Department Computer Science and Engineering, IIT Madras: http//nptel.ac.in/courses/106106139/
2. Machine Learning, Stanford University: https://see.stanford.edu/Course/CS229

Title	**Principles of Computer Systems - I**	Course No.	AIDSLXXX
Department	CSE, AI & DS	Structure (L-T-P [C])	2–0–2 [3]
Offered for	B.Tech (AI)	Type	Compulsory
Prerequisite	Maths for Computing	Antirequisite	Computer Architecture

Objectives
The Instructor will:
1. Explain necessary layered abstraction of a system
2. Provide an understanding of basic concepts of several hardware components and design of computing components
3. Introduce language translation schemes

Learning Outcomes
The students will have the ability to:
1. Design combinational and sequential circuits for a set of problems
2. Explain the working principles of several components of a computer
3. Design algorithms for language translations.

Contents
Introduction: Layered architecture of a system, Hardware, Software, Hardware-software Interaction (2 Lectures)
Digital Circuit Design: Combinational Circuits, Combinational Analysis and Design, Sequential Circuits, Sequential Analysis and Design (10 Lectures)
Computer System Design: Instruction set Architecture, CPU, Performance analysis, Pipelining, Memory Hierarchy, I/O, ILP (10 Lectures)
Language Translation**:** Levels of language translation, Compiler, Semantic Analysis, Code generation (6 Lectures)

Laboratory
Digital Logic Circuits, Digital Hardware, VHDL state machines, Machine Language Design, Memory, CPU, Assembler, Parallel Programming.

Text Books
1. S. WARFORD (2017), Computer Systems, Jones and Bartlett Learning, 5th Edition.
2. R. BRYANT, D. O'HALLARON (2016), Computer Systems A Programmer's Perspective, Pearson, 3rd Edition.

Reference Books
1. Saltzer, J. and Kaashoek, F. (2009), *Control Systems - Principles & Design*, 3rd Edition, Morgan Kaufmann
2. John L. Hennesy, David A. Patterson (2017), Computer Architecture: A Quantitative Approach, 6th Edition, Morgan Kaufmann.

Online Material
NPTEL Course, Foundations of Computer Systems Design, https://nptel.ac.in/courses/106106197/

Title	**Principles of Computer Systems - II**	Course No.	AIDSLXXX
Department	CSE, AI & DS	Structure (L-T-P [C])	3–0–2 [4]
Offered for	B.Tech (AI & DS)	Type	Compulsory
Prerequisite	Principles of Computer Systems - I (Same Semester)	Antirequisite	Operating Systems, Computer Networks

Objectives
The Instructor will:
1. Explain necessary layered abstraction of a system
2. Provide an understanding of the fundamentals of operating systems and networking
3. Introduce the concepts of Virtualization

Learning Outcomes
The students will have the ability to:
1. Explain the working principles of operating systems
2. Design interconnected environments using basic networking protocols.
3. Apply concepts of Virtualization in a practical environment

Contents
Introduction: Layered architecture of a system, Kernel, Network-Operating System Interaction (4 Lectures)
Process Management: Process, process states, concurrent processes, inter-process communication, Synchronization, Deadlock (12 Lectures)
Storage management: Memory Allocation, Virtual Memory, File Management (9 Lectures)
Network Design:Working principles, Layered Architecture, IP Addressing, Protocols for Transport, and Application Layers. (12 Lectures)
Virtualization: Basics of Virtual Machines, Containers, Virtualization techniques, Cloud and Data Centers (5 Lectures)

Laboratory
Introduction to the Linux environment, Process Management, and Synchronization, Scheduling, Memory Management, Client-Server message passing, Internet protocols, IP addressing, peer to peer protocol, virtualization.

Text Books
1. S. WARFORD (2017), *Computer Systems*, Jones and Bartlett Learning, 5th Edition.
2. N.F. SCHNEIDEWIND, (2012), Computer, Network, Software, and Hardware Engineering with Applications, Wiley-IEEE Press.

Reference Books
1. A. SILBERSCHATZ, P.B. GALVIN, G. GAGNE (2013), Operating System Concepts, Wiley 8th Edition.
2. S. DAS (2017), UNIX Concepts and Applications, Tata McGraw-Hill.
3. J. KUROSE, K.ROSS (2016), Computer Networking: A Top-Down Approach, Pearson, 7th Edition.

Online Material
1. NPTEL Course: Operating System Fundamentals, https://nptel.ac.in/courses/106/105/106105214/
2. NPTEL Course: Computer Network and Internet Protocol, https://nptel.ac.in/courses/106/106/106106091/

Course Title	**Principles of Programming Languages**	Course No.	**CSXXX**
Department	Computer Science & Engineering	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B.Tech	Type	Compulsory
Prerequisites	Data Structures and Algorithms, Computer Architecture, Operating Systems	Antirequisite	None

Objectives
1. To introduce major programming paradigms, and principles and techniques involved in design and implementation of modern programming languages.
2. To introduce frameworks for reasoning about programming languages.

Learning Outcomes
The students are expected to have the ability to:
1. Identify the abstract syntax of any programming language
2. Analyze and critique/appreciate the inherent philosophy of different programming languages
3. Design small domain-specific languages/language features and implement them as interpreters or embeddings in another language.

Contents
Introduction: Study of principles of programming paradigms like imperative, functional, object-oriented and logic programming. (2 Lectures)
Syntax and semantics of programming languages: Syntax and semi-formal semantic specification using attribute grammar. (4 Lectures)
Imperative Programming: Location, Reference and expressions, Assignment and Control, Data types, Blocks, Procedures, Modules. (6 Lectures)
Object-oriented programming: Classes, Objects, Abstraction, Encapsulation, Inheritance and software reuse. (6
Lectures)
Functional programming: Function as first-class objects, Higher order functions, Polymorphic data types, Type checking, Type inferencing, Recursion, Lambda calculus. (6 Lectures)
Logic programming: Horn clauses, Unification, SLD resolution, Backtracking, Cuts. (6 Lectures)
Concurrent programming: Processes, Synchronization primitives, Safety and liveness properties, Multithreaded programs. (6 Lectures)
Case Studies: C++, Java, Haskell, Prolog, Python. (6 Lectures)

Text Books
R. SETHI (2006), Programming Languages: Concepts and Principles, Pearson, 2nd Edition.
T. W. PRATT, M.V. SELKOWITZ (2000), Programming Languages: Design and Implementation, Pearson, 4th Edition.

Reference Books
B.J. MacLENNAN (1999), Principles of Programming Languages: Design, Evaluation and Implementation, Oxford University Press, 3rd Edition.
M. BEN-ARI (2005), Principles of Concurrent and Distributed Programming, Pearson, 2nd Edition.
J.C. MITCHELL (1996), Foundations for Programming Languages, MIT Press.
B.C. PIERCE (2002), Types and Programming Languages, MIT Press, 1st Edition.
H.P. BARENDREGT (2014), The Lambda Calculus: Its Syntax and Semantics, North-Holland, 2nd Edition.
D.P. FRIEDMAN, M. WAND (2008), Essentials of Programming Languages, MIT Press, 3rd Edition.

Course Title	**Software Engineering**	Course No.	**CSL2XX**
Department	Computer Science & Engineering	Structure (L-T-P [C])	3-0-2 [4]
Offered for	B.Tech CSE	Type	Compulsory
Prerequisite	Introduction to Computer Science	Antirequisite	None

Objectives

1. This course focuses on engineering practices and processes that development team uses to make sure the team is setup for change
2. The course introduces practices, techniques and processes that can help team build high quality software

Learning Outcomes

Students are expected to have the ability to:

Comfortably and effectively participate in various techniques and processes for building scalable and high quality software.

Contents

Introduction of Software Engineering: Need for software engineering, Software quality attributes, Software product pipelines, Software life cycle models and processes, Requirement engineering using UML Diagrams. (9 Lectures)

Software Architecture and Design: Design principles, Design Patterns, Architecture Versus Design, Modularity, Software Components and Connectors, Architecture Styles. (6 Lectures)

Essence of Modern Software Engineering: Software engineering essence, Essence language, Essence kernel, Using essence kernel in agile development practices, Agile Principles, Agile process models through essence kernel, Large scale complex development Using kernel. (13 Lectures)

Software Testing: Quality metrics, Coding style and Static analysis tools, Verification and validation, Various testing techniques and Test case generations. (7 Lectures)

Software Project Management: Software versioning and Continuous integration, Project management and Risk analysis, Configuration management, Cost analysis and estimation. (7 Lectures)

Laboratory

Assignment/Project on Software requirement acquisition, UML diagrams, Preparing software requirement specification, Practicing agile methods, User story, Backlog, Test case generation, Unit testing, CI configurations, Cost estimation, Manpower management and Sprint analysis using Burn down charts.

Text Books

R.S. PRESSMAN, B.R. MAXIM (2019), Software Engineering: A Practitioner's Approach, McGraw-Hill India, 2019, 9th Edition.

Mark Richards, Neal Ford (2020), Fundamentals of Software Architecture, O'Reilly Media, Inc.

L. BASS, P. CLEMENTS, R. KAZMAN (2012), Software Architecture in Practice, Pearson, 3rd Edition.

I. JACOBSON, H. LAWSON, P.W. NG, P.E. McMAHON, M. GOEDICKE (2019), The Essentials of Modern Software Engineering, ACM Books.

300 - Level Courses

• Compulsory Courses (CSE and AI&DS)

Course Title	**Artificial Intelligence (300)**	Course No.	**CSL3xx**
Department	CSE, AI&DS	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B.Tech (CSE, AI&DS)	Type	Compulsory
Prerequisite	Data Structures and Algorithms	Antirequisite	Artificial Intelligence (700) - CSL7xx

Objectives
To provide the foundations for AI problem solving techniques and knowledge representation formalisms.

Learning Outcomes
The students are expected to have the ability to:
1. Identify and formulate appropriate AI methods for solving a problem.
2. Implement AI algorithms
3. Compare different AI algorithms in terms of design issues, computational complexity, and assumptions

Contents
Introduction: Uninformed search strategies**,** Greedy best-first search, And-Or search, Uniform cost search, A* search, Memory-bounded heuristic search (5 Lectures)
Local Search Techniques: Beam Search, Hill Climbing Search, Genetic Search techniques (2 lectures)
Constraint Satisfaction Problems: Backtracking search for CSPs, Local search for CSPs (4 Lectures)
Adversarial Search: Optimal Decision in Games, The minimax algorithm, Alpha-Beta pruning, Expectimax search (5 Lectures)
Knowledge and Reasoning: Propositional Logic, Reasoning Patterns in propositional logic; First order logic: syntax, semantics, Inference in First order logic, unification and lifting, backward chaining, resolution (7 Lectures)
Representation: Information extraction, representation techniques, foundations of Ontology (4 Lectures)
Planning: Situation Calculus, Deductive planning, STRIPES, sub-goal, Partial order planner (4 Lectures)
Bayesian Network and causality: Probabilistic models, directed and undirected models, inferencing, reasoning, causality (6 lectures)
Reinforcement Learning: MDP, Policy, Q-value (5 Lectures)

Text Book
S. RUSSEL, P. NORVIG (2020), Artificial Intelligence: A Modern Approach, Pearson, 4th Edition.

Reference Books
1. E. RICH, K. KNIGHT, S. B. NAIR (2017), Artificial Intelligence, McGraw Hill Education, 3rd Edition.
2. J. PEARL (2009), Causality: Models, Reasoning and Inference, Cambridge University Press, 2nd Edition.
3. D. KOLLER, N. FRIEDMAN (2009), *Probabilistic Graphical Models: Principles and Techniques,* MIT Press

Course Title	**Computer Architecture**	Course No.	**CSL3XX / EELXXX**
Department	CSE and EE	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B. Tech. CSE	Type	Compulsory
Prerequisite	None	Antirequisite	

Objectives

1. To understand aspects of computer architecture and program performance
2. To provide essential understanding of different subsystems of modern computer system and design aspects these subsystems
3. To introduce hands-on experience of computer architecture design and performance enhancement

Learning Outcomes

The students are expected to have the ability to:

1. Identify basic components and design of a computer, including CPU, memories, and input/output units
2. Identify issues involved in the instruction execution and various stages of instruction life stage
3. Identify issues related to performance improvement

Contents

Introduction: Basic computer organization, Components of computer systems, information representation. (3 Lectures)

Central Processing Unit: Arithmetic and Logic Unit; Instruction sets; RISC, CISC, and ASIC/ASIP paradigms; Various addressing modes; Assembly language programming; Instruction interpretation: micro-operations and their RTL specification; CPU design, Hardwired and microprogrammed, Performance issues: Parallel processing, Pipelining, Hazards, Advanced parallelization techniques. Cache Coherence protocols, Multicore Architecture (16 Lectures)

Memory Hierarchy: Memory organization, Various levels of memory architecture and their working principles, Cache memory, Writing strategy, Coherence, Performance issues and enhancement techniques for memory design. (14 Lectures)

Interfacing: I/O transfer techniques: Program controlled, Interrupt controlled and DMA; Introduction to computer buses, Peripherals and current trends in architecture. (9 Lectures)

Text Books

D.A. PATTERSON, J.L. HENNESSY (2008), Computer Organization and Design, Morgan Kaufmann, 4th Edition.

W. STALLINGS (2015), Computer Organization and Architecture: Designing for Performance, Pearson Education India, 10th Edition.

References

A.S. TANENBAUM (2013), Structured Computer Organization, Prentice Hall of India, 6th Edition.

V.C. HAMACHER, Z.G. VRANESIC, S.G. ZAKY (2011), Computer Organization, McGraw Hill, 5th Edition.

J.L. HENNESSY, D.A. PATTERSON (2017), Computer Architecture: A Quantitative Approach, Morgan Kaufmann, 6th Edition.

D.V. HALL, S.S.S.P. RAO (2017), Microprocessors and Interfacing, McGraw Hall, 3rd Edition.

Course Title	**Computer Networks**	Course No.	**CSL3xx**
Department	Computer Science and Engineering	Structure (L-T-P [C])	3-0-3 [4.5]
Offered for	B.Tech	Type	Compulsory
Prerequisite	Prob. Stat. and Stochastic Process	Antirequisite	None

Objectives
1. To understand the organization of computer networks, factors influencing the performance of computer networks, and the reasons for having a variety of different types of networks.
2. To understand the Internet structure, various protocols of the Internet and how these protocols address the standard problems of networking and the Internet.
3. Hands-on experience on networking fundamentals through practical sessions.

Learning Outcomes
Students are expected to have the ability to:
1. Explain essential protocols of computer networks in terms of design, implementation & operations.
2. Identify various design parameters such as latency, bandwidth, error rate, throughput, and their influence on node/link utilization and performance

Contents
Introduction: Layer approach, Packet switching techniques, Performance metrics delay, loss, throughput, bandwidth delay product, latency, Basic understanding of physical and data link layer. (8 Lectures)
Network Layer: Internetworking, Tunneling, Encapsulation, Fragmentation, Internet protocol and its operation, Routing algorithms distance vector and link state algorithm and Routing protocols. (8 Lectures)
Transport Layer: Transmission Control Protocol, Flow control, Error control, Congestion control, Header, Services, Connection management, Timers, Congestion control; User Datagram Protocol, Introduce low latency
protocols (WebRTC, LHLS), Applications (IoT and FOG, Real-time applications, ad-hoc wireless protocols). (10 Lectures)
Applications: Network programming, Socket abstraction, Peer-to-peer architecture (P2P architecture), Client
server architecture, DNS, HTTP, FTP, SMTP, TelNet, etc. (7 Lectures)
Advanced Internetworking: Multicast routing, Queuing disciplines and buffer management techniques. (6 Lectures)
Network security: Public key and private key cryptography, Digital signature, Firewalls. (3 Lectures)

Laboratory
Networking hardware (Understanding cables, switches, routers, Setting up switching network, Setting up subnets and routing across the subnets)
Socket programming - Development of client-server applications using sockets (possible examples, file transfer, peer-peer applications, chat, network monitor etc.)
Networking commands - ifconfig, route, arp, arping, ping, netstat. tcpdump, host, nslookup, dig, ftp, scp, ssh, finger, dhclient, dhcrelay etc.
Protocol analyzer - closely looking at protocols (HTTP, TCP, UDP, ICMP, 802. 3, DHCP, DNS etc.) headers and analyzing the interactions between client and server of different applications
QualNet simulator/Packet Tracer (Implementation of ARQs - Stop-and-wait, Sliding Window Go Back N etc. Verifying operations of routing protocols, influence of congestion on end users performance, congestion control algorithms Reno, New Reno, Cubic, router buffer size on end users performance.

Text Book
K.W. ROSS, J.F. KUROSE (2016), Computer Networks: A Top Down Approach, Pearson, 7th Edition.

Reference Books
W. STALLINGS (2013), Data and Computer Communications, Pearson, 10th Edition.
L.L. PETERSON, B.S. DAVIE (2020), Computer Networks: A Systems Approach, Morgan Kaufmann, 6th Edition.

Course Title	**Database Systems**	Course No.	**CSL3XXX**
Department	Computer Science and Engineering	Structure (L-T-P [C])	3-0-2 [4]
Offered for	B.Tech	Type	Compulsory
Prerequisite	Data Structures and Algorithms	Antirequisite	None

Objectives

1. To understand concepts of database design, database management systems and their applications, data modeling, and query languages.
2. To understand different file-structures, transaction management schemes, concurrency control processes, database recovery mechanisms, query processing and optimization.

Learning Outcomes

Students are expected to have the ability to:

1. Apply different data modeling methods to requirement analysis, design, and implementation of database systems.
2. Deduce normalized forms for efficient relational database design.
3. Use appropriate storage and access structures.
4. Use techniques for transaction management, concurrency control and recovery.
5. Analyze complexity issues of query execution.

Contents

Preliminaries: Database system concepts and architecture. (2 Lectures)
Semantic Data Model: Entity-Relationship (ER) model, Enhanced Entity-Relationship (EER) model. (4 Lectures)
Relational Model and Languages: Relational data model, Relational database constraints, Relational database
design by ER- and EER-to-relational mapping, Relational algebra and Relational calculus. (9 Lectures)
SQL: Schema definition, Constraints, Queries, Views. (5 Lectures)
Database Design Theory: Functional dependencies, Normalization. (5 Lectures)
Algorithms for Query Processing and Query Optimization (5 Lectures)
Database File Organization: Disk storage, File structures, Hashing techniques, Indexing structures. (3 Lectures)
Database Transaction Processing: Transaction concepts and theory, Concurrency control protocols, Database
recovery protocols. (6 Lectures)
Advanced Databases: Distributed databases, Object-oriented databases, Multimedia databases, Handling unstructured data, Big data, NoSQL databases. (3 Lectures)

Laboratory

Exercises in SQL covering: Data definition, Data manipulation, Data retrieval, Transaction control
Exercises in PL/SQL covering procedures, functions, loops, triggers
NoSQL data organization and query optimization.

Text Book

R. ELMASRI, S.B. NAVATHE (2017), Fundamentals of Database Systems, Pearson Education, 7th Edition.

Reference Books

1. R. RAMAKRISHNAN, J. GEHRKE (2014), Database Management Systems, McGraw-Hill, 3rd Edition.
2. H.G. MOLINA, J. ULLMAN, J. WIDOM (2014), Database Systems: The Complete Book, Pearson, 2nd Edition.
3. P. RAJ, A. RAMAN, D. NAGARAJ, S. DUGGIRALA (2015), High-Performance Big-Data Analytics: Computing Systems and Approaches, Springer, 1st Edition.
4. N. SABHARWAL, S. GUPTA EDWARD (2014), Big Data NoSQL: Architecting MongoDB, CreateSpace Independent Publishing Platform, 1st Edition.
5. K. BANKER, P. BAKKUM, S. VERCH, D. GARRETT, T. HAWKINS (2016), MongoDB in Action, Manning Publications, 2nd Edition.
6. G. HARRISON (2015), Next Generation Databases: NoSQL and Big Data, Apress, 1st Edition.

Title	**Data Engineering**	Course No.	CSLXXX
Department	Computer Science and Engineering, Maths	Structure (L-T-P [C])	3-0-3 [3]
Offered for	B.Tech AI&DS	Type	Compulsory
Prerequisite	Principles of Computer System - II/ Operating Systems	Antirequisite	Database Systems

Objectives
The Instructor will:
1. Introduce fundamental concepts in representing data, accessing it and analysing it
2. Explore applications in data science and big data projects

Learning Outcomes
The students will have the ability to:
1. Develop suitable data science ecosystem for the given application
2. Understand various data storage and retrieval techniques
3. Understand SQL and NoSQL databases and their usage
4. Analyse data using Python and Python-based tools

Contents
Introduction: Data source, Big Data, Structured and unstructured data (2 Lectures)
Data Models and Storage: Relational databases, NoSQL database, normalized and denormalized data models, Data cleaning, Distributed Data Storage and Management, Hashing, Indexing (14 Lectures)
Query processing: Querying big data using SQL and NoSQL, Elastic Search, Query optimization, speeding up, maintaining ACID property, Design Patterns, Data reliability, quality and provenance, Distributed query processing, Query optimization and Processing (16 Lectures)
Data Warehousing: OLAP, OLTP (4 Lectures)
Streaming Data analytics: In-memory Analytics, data pipelines and dashboards, Predictive Analytics (6 Lectures)

Laboratory
Lab exercises should be in accordance with the theory Lectures. The lab sessions may cover the following topics:
1. Data Collection Techniques
2. Indexing implementation for the structured data and unstructured data
3. SQL queries (schema, DDL, DML, DQL)
4. Data format interchange using XML, JSON
5. NoSQL system (HBase, Hive, MongoDB)
6. Data wrangling, data operations (e.g. NumPy)
7. Hadoop, Spark and MapReduce

Textbooks
1. M. KLEPPMANN (2017), Designing Data-Intensive Applications The Big Ideas Behind Reliable, Scalable, and Maintainable Systems, O'Reilly.
2. L. WEISE (2015), Advanced Data Management: For SQL, NoSQL, Cloud and Distributed Databases, Walter de Gruyter GmbH.
3. A. SILBERSCHATZ, H.F. KORTH, S. SUDARSHAN (2011), Database System Concepts, McGraw Hill Publications, 6th Edition.

Reference books
1. H.G. MOLINA, J. ULLMAN, J. WIDOM (2014), Database Systems: The Complete Book, Pearson, 2nd Edition.
2. P. RAJ, A. RAMAN, D. NAGARAJ, S. DUGGIRALA (2015), High-Performance Big-Data Analytics: Computing Systems and Approaches, Springer, 1st Edition.

Self-Learning Material

1. NPTEL course on 'Indexing and Searching Techniques in Databases' by Dr Arnab Bhattacharya, IIT Kanpur: https://nptel.ac.in/courses/106/104/106104021/
2. NPTEL course on 'NOC: Fundamentals of Database Systems' by Dr Arnab Bhattacharya, IIT Kanpur: https://nptel.ac.in/courses/106/104/106104135/

Course Title	**Maths for Big Data**	Course No.	MAL3XXX
Department	Mathematics	Structure (L-T-P [C])	2-1-0 [3]
Offered for	B.Tech (AI & DS)	Type	Compulsory
Prerequisite	Mathematics - II, Probability, Statistics and Stochastic Processes	Antirequisite	None

Objectives

To introduce basic concepts and core techniques which enable the students to handle large data and equip students with sufficient knowledge of core techniques which can be used by the students in their respective fields of interest.

Learning Outcomes

Students will have the ability to:

1. Understand novel techniques to handle a large amount of data.
2. Develop concepts and tools to ingest, process and analyse massive data in real time.

Contents

Statistical Methods: Multivariate Analysis, Sampling theory: simple random sampling, stratified sampling, cluster sampling, ratio and regression estimators, two stage sampling, Compressive sensing. (14 Lectures)

Numerical linear algebra: Spectral decomposition, Schur Decomposition, QR Factorization, Singular value decomposition (SVD), PCA for large matrices, Algorithms for big matrices, Least square approximations, Low-Rank Approximation, Manifolds. (14 Lectures)

Text Books

1. D. MINER, A. SHOOK (2016), Mapreduce Design Patterns: Building Effective Algorithms and Analytics for Hadoop and Other Systems, O'Reilly Media.
2. V. MAYER-SCHÖNBERGER, K. CUKIER (2013), Big Data: A Revolution that Will Transform How We Live, Work, and Think, Houghton Mifflin Harcourt.
3. N. MARZ, J. WARREN (2015), Big Data: Principles and Best Practices of Scalable Real-time Data Systems, Manning.

Reference Books

1. M. MITTAL, V.E. BALAS, D.J. HEMANTH, R. KUMAR (2018), Data Intensive Computing Applications for Big Data, IOS Press.
2. B. FURHT, A. ESCALANTE, (Eds.), Handbook of Data Intensive Computing, Springer, 2011, 1st Edition.
3. G. STRANG (2005), Linear Algebra and its Applications, Cengage Learning, 4th Edition.

Course Title	**Operating Systems**	Course No.	**CS 3XXX**
Department	Computer Science and Engineering	Structure (L-T-P [C])	3-0-2 [4]
Offered for	B.Tech	Type	Compulsory
Prerequisite	Data Structures and Algorithms	Antirequisite	None

Objectives
1. To learn about design principles of operating systems
2. To learn various functionalities and services of an operating system in general
3. To do a case study of Operating System

Learning Outcomes
1. Ability to understand the system performance that is driven by the Operating System services
2. Ability to modify and compile OS
3. Ability to diagnose and solve various problems in Operating Systems

Contents
Overview of Operating Systems*:* Types of Operating Systems, System calls and OS structure. (4 Lectures)
Processes Management*:* Process, Threads, CPU scheduling. (6 Lectures)
Process Coordination*:* Mutual exclusion, Mutex implementation, Semaphores, Monitors and condition variables, Deadlocks. (10 Lectures)
Memory Management*:* Swapping, Paging, Segmentation, Virtual memory, Demand paging, Page Replacement algorithms. (8 Lectures)
Storage Management*:* I/O devices and drivers, Disks and file Systems, File layout and directories, File system performance, File system reliability. (8 Lectures)
Protection and Security*:* System protection, System security. (6 Lectures)

Laboratory
Designing a shell in Linux
Multithreaded programming using
pthread Solving the Sleeping-Barber
problem Modification of scheduling
algorithm in Linux
Solving the Producer-Consumer problem over a
network Finding text, data, and stack segments of a
process in Linux Implementation of page
replacement algorithms
Changing file attributes in Linux
Implementing an encrypted file system in Linux
Implementing symbolic links in Linux

Text Book
A. SILBERSCHATZ, P.B. GALVIN, G. GAGNE (2018), Operating System Concepts, John Wiley & Sons Inc., 10th Edition.

Reference Book
A.S. TANENBAUM, A.S. WOODHULL (2006), Operating Systems Design and Implementation, Pearson, 3rd Edition.
W. STALLINGS (2017), Operating Systems Internals and Design Principles, Pearson, 9th Edition.

400 - Level Courses

• Compulsory Courses (CSE and AI&DS)

Course Title	**Data Visualization (400)**	Course No.	CSL4xx
Department	Maths, CSE	Structure (L-T-P [C])	3-0-3 [4.5]
Offered for	B.Tech (AI&DS)	Type	Compulsory
Prerequisite	Data Engineering / Database Systems	Antirequisite	Data visualization (700) - CSL7xx

Objectives
The instructor will
1. Explain techniques and algorithms for creating effective visualizations based on principles from graphic design.
2. Introduce several industry-standard software tools to create a compelling and interactive visualization of various types of data.

Learning Outcomes
1. An understanding of the key techniques and theory used in visualization, including data models, graphical perception, and techniques for visual encoding and interaction.
2. Exposure to a number of common data domains and corresponding analysis tasks, including multivariate data, networks, text, and cartography.
3. Practical experience building and evaluating visualization systems.

Contents
Introduction: Data for Graphics, Design principles, Value for visualization, Categorical, time series, and statistical data graphics, Introduction to Visualization Tools (3 Lectures)
Graphics Pipeline: Introduction, Primitives: vertices, edges, triangles, Model transforms: translations, rotations, scaling, View transform, Perspective transform, window transform (3 Lectures).
Aesthetics and Perception: Graphical Perception Theory, Experimentation, and the Application, Graphical Integrity, Layering and Separation, Color and Information, Using Space Effectively (5 Lectures)
Visualization Design: Visual Display of Quantitative Information, Data-Ink Maximization, Graphical Design, Exploratory Data Analysis, Heat Map (8 Lectures)
Multidimensional Data: Query, Analysis and Visualization of Multi-dimensional Relational Databases, Interactive Exploration, tSNE (5 Lectures)
Interaction: Interactive Dynamics for Visual Analysis, Visual Queries, Finding Patterns in Time Series Data,
Trend visualization, Animation, Dashboard, Visual Storytelling (8 Lectures)
Collaboration: Graph Visualization and Navigation, Online Social Networks, Social Data Analysis, Collaborative Visual Analytics, Text, Map, Geospatial data (10 Lectures)

Laboratory
Visualization Design, Exploratory data analysis, Interactive Visualization Tools like Tableau, Gephi, D3, etc. Mini Project.

Text Books
1. E. TUFTE (2001), The Visual Display of Quantitative Information, Graphics Press, 2nd Edition.
2. J. KOPONEN, J. HILDÉN (2019), Data Visualization Handbook, CRC Press.

Reference Books
1. M. LIMA (2014), The Book of Trees: Visualizing Branches of Knowledge, Princeton Architectural Press.
2. R. TAMASSIA (2013), Handbook of Graph Drawing and Visualization, CRC Press.
3. S. MURRAY (2017), Interactive Data Visualization for the Web, O'Reilly Press, 2nd Edition.

400 - Level Compulsory Course

Course Title	**Deep Learning (400)**	Course No.	**CSL4xx**
Department	Computer Science and Engineering, AI&DS	Structure (L-T-P [C])	3-0-3 [4.5]
Offered for	B.Tech.	Type	Compulsory
Prerequisite	Introduction to Machine Learning/Pattern Recognition and Machine Learning	Antirequisite	Deep Learning (700) - CSL7xx

Objectives
The objective of this course is
1. To introduce students through some of the latest techniques in deep learning.
2. Hands on and the students should be able to design intelligent deep learning systems for solving the problems in the area of their interests.

Learning Outcomes
The students are expected to have the ability to:
1. Understand various deep learning models such CNN, Autoencoders, RNN etc.
2. Analyze various applications solved through the use of deep learning models
3. Design and implement their own deep learning models for the problem of their choice

Contents
Neural networks: DL Optimizers (SGD, MBGD, AdaGrad, Adam) and Regularization, Initialization Methods (7 Lectures)
DL Models: Autoencoder, Convolutional Neural Networks, Recurrent Neural Networks, LSTM, Network Architecture Search (NAS) (14 Lectures)
Deep Generative Models: Deep Belief Networks, Variational Autoencoders, Generative Adversarial Networks, Deep Convolutional GAN (12 Lectures)
Representation learning: Unsupervised Pre-training, Transfer learning and Domain adaptation, Distributed representation, Discovering underlying causes (9 Lectures)

Laboratory
Autoencoder, CNN, LSTM, DBM, GANs (variants), Transfer Learning, NLM, Graph NN, Adversarial losses

Text Book
I. GOODFELLOW, Y. BENGIO, A. COURVILLE (2016), Deep Learning, The MIT Press, 1st Edition.

Reference Books
1. A. ZHANG, Z. LIPTON, M. LI, A. SMOLA (2020) Dive into Deep Learning (Release 0.7.1), https://d2l.ai/d2l-en.pdf.
2. D. FOSTER (2019), Generative Deep Learning, O'Reilly Media, 1st Edition.

Self-learning Material
1. Practical Machine Learning with Tensorflow, NPTEL Course Material, Department Computer Science and Engineering, IIT Madras: https://nptel.ac.in/courses/106106213/
2. Stanford CS class (CS231n), Convolutional Neural Networks for Visual Recognition: http://cs231n.github.io/

Course Title	**Dependable AI (400)**	Course No.	**CSL4xx**
Department	CSE, AI&DE	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B.Tech (AI&DS)	Type	Compulsory
Prerequisite	Introduction to Machine Learning/ Pattern Recognition and Machine Learning, AI	Preferred Knowledge / Antirequisite	Preferred: Deep Learning (same semester) Antirequisite: Dependable AI (700)

Objectives
The Instructor will:
Provide characteristic details of AI and machine learning systems to make them dependable, such as explainability, interpretability, safety etc.

Learning Outcomes
Students are expected to have the ability to:
1. Assess the dependability of AI systems.
2. Develop explainable, robust, and safe AI models.

Contents
Introduction: Overview, Motivation, Challenges – medical and surveillance (3 Lectures)
Explainable AI: Accuracy-explainabilityTradeoff, Interpretability Problem, Predictability, Transparency, Traceability, Causality, Reasoning, Attention and Saliency (10 Lectures)
Interpretable AI: Prediction Consistency, Application Level Evaluation, Human Level Evaluation, Function Level Evaluation (5 Lectures)
Adversarial Robustness: Adversarial Attacks and Defenses (8 Lectures)
Trustworthy AI: Integrity, Reproducibility, Accountability (2 Lectures)
Bias-free AI: Accessibility, Fair, Data Agnostics Design, Disentanglement (4 Lectures)
Privacy Preserving AI: Federated Learning, Differential Privacy and Encrypted Computation (6 Lectures)
Verified AI: Environment and Specification Modeling, Design with Formal Inductive Synthesis, Evaluation Platforms for AI Safety (4 Lectures)

Textbooks
1. J. PEARL (2018), The Book of Why: The New Science of Cause and Effect, Basic Books.
2. N. BOSTROM (2014), The Ethics of Artificial Intelligence. The Cambridge Handbook of Artificial Intelligence, Cambridge University Press.

Self-learning Material
Udacity course on Secure and Private AI: https://www.udacity.com/course/secure-and-private-ai--ud185

Course Title	**Optimization in ML**	Course No.	**CSL4xx**
Department	Mathematics	Structure (L-T-P [C])	3-0-3 [4.5]
Offered for	B.Tech (AI&DS)	Type	Compulsory
Prerequisite	PRML or Introduction to ML	Antirequisite	Optimization

Objectives
1. To understand the theory of optimization methods and algorithms developed for solving various types of optimization problems
2. To apply the mathematical results and numerical techniques of optimization theory to Machine Learning problems

Learning Outcomes
Students are expected to have the ability to:
1. Compose existing theoretical analysis with new aspects and algorithm variants.
2. Formulate the most important optimization algorithms for machine learning applications

Contents
Introduction to optimization: Machine Learning and Optimization, linear and non-linear optimization, discrete optimization, Network flows, convex sets, functions. (14 Lectures)
Regularizations and SGD: L1 and L2 regularization, First-order methods: gradient descent, acceleration and subgradient method, Stochastic gradient methods, SGD heuristics and tricks, escaping saddle points. (14 Lectures)
Other topics relevant to optimization for ML: Interior point and cutting-plane methods for ML, Min-Max Problems (convex-concave and nonconvex),Non-Euclidean and Submodular optimization. (14 Lectures)

Reference Books
1. S. BOYD, L. VANDENBERGHE (2003), Convex Optimization, Cambridge University Press.
2. S. SRA, S. NOWOZIN, S. WRIGHT, (Eds.), Optimization for Machine Learning, MIT Press, 2011.
3. E. HAZAN (2019), Lecture Notes: Optimization for Machine Learning [https://arxiv.org/abs/1909.03550].
4. D. BERTSEKAS (2016), Nonlinear Programming, Athena Scientific, 3rd Edition.

Self Learning Material
1. http://suvrit.de/teach/6881/

Title	**Theory of Computation**	Course No.	CSL4xxx
Department	Computer Science and Engineering	Structure (L-T-P [C])	
Offered for	B.Tech (CSE)	Type	Elective
Prerequisite	Maths for Computing	Antirequisite	

Objectives
To learn about computability techniques
To learn about advanced computational complexity models

Learning Outcomes
1. To be able to distinguish between computable and uncomputable problems
2. Gaining in-depth understanding of advanced complexity models

Contents
Computability: Review of Turing Machine, NP and NP-completeness, Diagonalization, view of PDAs, 2DFAs, FAs as restricted TMs, and related theorems. Tape reduction, and robustness of the model. Encoding and Enumeration of Turing Machines, Undecidability. Rice-Myhill-Shapiro theorem. Relativisation.
Decision Trees and Communication Complexity:
Certificate Complexity, Randomized Decision Trees, Lower bounds on Randomized Complexity, Some techniques for decision tree lower bounds, Comparison trees, and sorting lower bounds, Yao's MinMax Lemma, Definition of communication complexity, Lower bound methods, Overview of other communication models, Applications of communication complexity.
Time Complexity: Time as a resource, Linear Speedup theorem, Crossing sequences and their applications, Hierarchy theorems. P vs NP. Time Complexity classes and their relationships. Notion of completeness, reductions. Cook-Levin Theorem. Ladner's theorem. Relativization Barrier: Baker-Gill-Solovay theorem.
Space Complexity: Space as a resource. PSPACE, L and NL. Reachability Problem, Completeness results. Savitch's theorem, Inductive Counting to show the Immerman-Szelepcsenyi theorem. Reachability problems, Expander Graphs, SL=L
Complexity of Counting & Randomization: Counting problems, Theory of #P-completeness. The complexity classes PP, ParityP, BPP, RP, BPP are in P/poly, Toda's theorem.

Textbooks
D. Kozen (2013), Automata and Computability, Springer
D. Kozen (2006), Theory of Computation, Springer
S. Arora and B. Barak (2009), Complexity Theory: A Modern Approach, Cambridge University Press.

Reference books
Sipser, M., (2013), Introduction to the Theory of Computation, Cengage Learning
Hopcroft,J. E., Motwani, R., and Ullman, J. D., (2007), Introduction to Automata Theory, Languages, and Computation, Pearson

400 - Level Courses

- ## Elective Courses (CSE and AI&DS)

Course Title	**Advanced Data Structures**	Course No.	**CS4xxx**
Department	Computer Science and Engineering	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B.Tech CSE, AI&DS	Type	Elective
Prerequisite	Data Structures and Algorithms	Antirequisite	None

Objectives
Data structures are one of the important elements of modern computer science. The objective of the course is to learn major results and current research directions in data structure beyond the traditional linear and non-linear data structures.

Learning Outcomes
Students will have the ability to:
1. Use unconventional and modern data structures for the changing need of new age algorithmic problems.
2. Solve problems using data structures.

Contents
Temporal Data Structures: Persistent and retroactive data structures, Pointer machine. (5 Lectures)
Geometric Data Structures: Point location, Range searching, Fractional cascading, Data structures for moving data. (5 Lectures)
Dynamic optimality: Analytic bounds, Splay trees, Geometric view, independent rectangle, key-independent optimality. (5 Lectures)
External Memory Data Structures: Cache-oblivious (CO) Data structures such as B-Tree, CO priority Queue etc. (5 Lectures)
Integer Data Structures: Predecessor/Successor problems, van Emde Boas, x-fast and y-fast trees, Fusion trees: Sketching, Parallel comparison, Most significant set bit. (6 Lectures)
Static Trees and String Data Structures: Least common ancestor, Range minimum queries, Level ancestor, Suffix tree, Suffix array, Linear-time construction for large alphabets, Suffix tray, Document retrieval. (5 Lectures)
Memory Efficient Data Structures: Rank, Select, Tries, Compact suffix arrays and trees. (5 Lectures)
Dynamic Graphs: Link-cut trees, Heavy-light decomposition, Euler tour trees, Decremental connectivity in trees, Fully dynamic connectivity. (6 Lectures)

Reference Books
R.E. TARJAN (1983), Data Structures and Network Algorithms, SIAM. (covers BSTs, splay trees, link-cut trees)
P. MORIN (2013), Open Data Structures, Athabasca University Press, 31st Edition. (covers BSTs, B-trees, hashing, and some integer data structures)

Self Learning Material
MIT Lectures in Advanced Data Structures (6.851): https://courses.csail.mit.edu/6.851/fall17/lectures/

Title	**Computational Cognition & Behavior Modeling**	Number	CS L4XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech (All)	Type	Elective
Prerequisite	Probability & Statistics, PRML	Antirequisite	None

Objectives

The Instructor will:

1. Draw on formal models from classic and contemporary artificial intelligence to explore fundamental issues in human cognition and behavior.
2. Encourage students to explore answers to questions on cognition and behavior (e.g., What are the different forms that our knowledge of the world takes? How does this lead to behaviour?) and ways to model them
3. Help students develop an understanding of how computational modeling can advance cognitive science, how cognitive science can contribute to research in machine learning and AI, and how to fit and evaluate cognitive models to understand behavioral data.

Learning Outcomes

The students will have the ability to:

1. Demonstrate knowledge of basic concepts and methodologies of cognitive and behavioral modelling - be able to design simple models for sample problems.
2. Demonstrate understanding of the relationship between computational models and psychological theories - critically assess the psychological adequacy of a given model.
3. Qualitatively and quantitatively evaluate computational models of cognition and behavior.
4. Extend the understanding of these models to the design of architectures for social cognitive agents for applications such as autonomous driving, and assistive robotics.

Contents

Fractal 1 [Computational Modelling Techniques for Cognition and Behavior]

Introduction [1 Lecture]: The motivation underlying the computational modelling of cognition, and the possible questions that can be answered

Probabilistic approaches to modelling cognition & behavior [4 lectures]: Bayesian inference, hierarchical Bayesian models, probabilistic graphical models

Inductive Logic Programming & Language of thought models. *[3 Lectures]* Deep learning, reinforcement learning for modelling cognition. *[3 Lectures]* Information sampling & Active learning. *[3 Lectures]*

Fractal 2 [Cognition Modelling]

The mind as an information-processing system - Marr's levels of analysis (computational, algorithmic, implementation) [*1 Lecture*]

Structure and formation of intuitive theories of cognition of physical, biological and social systems; Learning causal relations. *[3 Lectures]*

Symbolic (rule-based), subsymbolic (probabilistic) & connectionist (network-based) models of cognition [*3 Lectures*]

Methodology and issues in the development and evaluation of cognitive models [7 Lectures]: Which psychological data are relevant? How do we collect such data (e.g., EEG, Eye-tracker; across ages & genders)? What predictions can be made by the models of cognition? How can these predictions be tested? - Some possible example models: language processing, memory, reasoning, categorization.

Fractal 3 [Behavioral Modelling]

Introduction [*4 Lectures*]: Brain & behavior; Understanding social behavior; General motivation underlying computational modelling of behavior

Symbolic (rule-based), subsymbolic (probabilistic) & connectionist (network-based) approaches to behavior modelling [*3 Lectures*]

Methodology and issues in the development and evaluation of behavioral models [7 Lectures]: Which behavioral data are relevant? How do we collect such data (e.g., EEG, Eye-tracker; across ages, genders & societies)? What predictions can be made? How do these predictions apply to assistive system design?

Text Books

1. Russell & Norving (2020). Artificial Intelligence: A Modern Approach. 4th Edition. Prentice Hall.
2. Farrell & Lewandowsky (2018). Computational Modeling of Cognition and Behaviour. Cambridge University Press.

Reference Material

1. Issa and Isaias (2015). Sustainable Design: HCI, Usability and Environmental Concerns. Springer-Verlag.
2. McClelland & Rogers (2003). The Parallel Distributed Processing Approach to Semantic Cognition. Nature Reviews Neuroscience. 4(4), 310-322.
3. Peterson, Abbott & Griffiths (2016). Adapting Deep Network Features to Capture Psychological Representations. 38th Annual Conference of the Cognitive Science Society.
4. Tenenbaum, Kemp, Griffiths & Goodman (2015). Rational Use of Cognitive Resources: Levels of Analysis between the Computational and the Algorithmic. Topics in Cognitive Sciences. 7(2), 217-229.
5. Wilson & Collins (2019). Ten Simple Rules for the Computational Modelling of Behavioral Data. eLife. 8:e49547.
6. Goodman, Tenenbaum & Gerstenberg (2014). Concepts in a Probabilistic Language of Thought. Center for Brains, Minds and Machines (CBMM).

Course Title	**Distributed Systems**	Course No.	CS4XX
Department	Computer Science and Engineering	Structure (L-T-P [C])	2-0-0 [2]
Offered for	B.Tech	Type	Elective
Prerequisite	Operating Systems	Antirequisite	None

Objectives
To introduce issues, challenges and approaches related to distributed systems

Learning Outcomes
Students are expected to have the ability in:
1. Designing, deploying and managing distributed systems
2. Understanding and analyzing various distributed algorithms and approaches

Contents
Introduction: Introduction to advanced operating systems and distributed systems. (2 Lectures)
Concurrency, Ordering, Races: Implementing Remote Procedure Calls, Time, Clocks, and the Ordering of events
in a distributed system, Distributed snapshots: Determining global states of distributed systems, Detecting concurrency bugs: Eraser & TSVD. (8 Lectures)
File Systems and Disks: A Fast file System for UNIX, Scale and performance in a distributed file system, The
Design and implementation of a log-structured file system, A case for redundant arrays of inexpensive disks (RAID). (6 Lectures)
Fault Tolerance*:* Implementing fault-tolerant services using the state machine approach, Paxos, Practical
Byzantine fault tolerance. (6 Lectures)
OS Kernels and Virtual Machines: Microkernels, System code verification, Kernels for multicore & disaggregation, Virtual machines. (6 Lectures)

Text Books
T. ANDERSON, M.DAHLIN (2014), Operating Systems: Principles and Practice, Recursive Books, 2nd Edition.
N. LYNCH (2009), Distributed Algorithms, Elsevier India.

Reference Books
W.R. STEVENS, S.A.RAGO (2013), Advanced Programming in the Unix Environment, Addison-Wesley, 3rd Edition.
W.R. STEVENS, B. FENNER, A.M. RUDOFF (2004), Unix Network Programming: Networking APIs: Sockets and XTI (Volume 1), Addison-Wesley, 3rd Edition.

Online References
http://pages.cs.wisc.edu/~remzi/Classes/537/Spring2018/
Https://www.cs.cmu.edu/~15712/syllabus.html

Title	**ICT for Development**	Number	CSL4XX0
Department	Computer Science and Engineering	L-T-P [C]	2-0-0 [2]
Offered for	B.Tech (All)	Type	Elective
Prerequisite	None	Antirequisite	None

Objectives
The Instructor will:
1. Facilitate an interactive platform for students to discuss and learn ICT strategies for various problems.
2. Reflect upon his/her own experiences in deploying ICT solutions in social settings and the associated challenges.

Learning Outcomes
Students will have the ability to:
1. Appreciate the potential that ICT possesses to make an impact on the society.
2. Learn through various case studies, that different problem settings require different solution approaches.

Contents
From the origins of ICTD to Big data and development [3 lectures]
Development theory and the critiques of ICTD [4 lectures]
Localization techniques and Identity management [3 lectures]
ICTD: knowledge economies and development [3 lectures]
e-learning and development [2 lectures]
e-governance and development [2 lectures]
ICTs and Environment sustainability [2 lectures]
Building sustainable rural information service networks [2
lectures] Sustainability failures of rural telecentres [1 lecture]
ICT for Agriculture [2 lecture]
ICTD: Information sharing practices [1 lecture]
ICTD for community networks and disaster management [2 lectures]
ICT4D: the future [1 lecture]

Text books
1. Heeks, R. (2017). *Information and communication technology for development (ICT4D)*. Routledge.

References
1. https://www.oii.ox.ac.uk/study/course/ICT_Development_Reading_List.pdf
2. http://act4d.iitd.ernet.in/act4d/index.php?option=com_content&view=article&id=33&Itemid=42
3. https://www.kth.se/student/kurser/kurs/kursplan/IV1009-20082.pdf?lang=en
4. https://www.manchester.ac.uk/study/masters/courses/list/06237/msc-icts-for-development/course-details/MGDI60701#course-unit-details

Course Title	**Introduction to Computer Graphics (400)**	Course No.	CSL4xx
Department	Computer Science and Engineering	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B.Tech, MTech, PhD	Type	Elective
Prerequisite	Data Structures and Algorithms	Antirequisite	Computer Graphics (700) - CSL7xx

Objectives
To provide a thorough introduction to computer graphics techniques, focusing on 2D and 3D modeling, image synthesis and rendering

Learning Outcomes
The students are expected to have the ability to:
1. Create and explain graphics primitives and interactive graphics applications in C++
2. Synthesize and render 2D and 3D worlds for visualization and animation

Contents
Introduction to Computer Graphics & Graphics Systems: Overview of computer graphics, representing pictures, preparing, presenting & interacting with pictures for presentations, OpenGL Primitives (3 Lectures)
Scan Conversions: Points & lines, Line drawing algorithms, Circle generation algorithm; Ellipse generating algorithm; scan line polygon, fill algorithm, Clipping algorithms: line and polygon, anti-aliasing (6 Lectures)
Transformations and Viewing: Basic transformations: translation, rotation, scaling; Matrix representations and homogeneous coordinates, transformations between coordinate systems; reflection shear; Viewing pipeline, Window to viewport coordinate transformation, clipping operations, viewport clipping, 3D viewing (9 Lectures)
Curves and Surfaces: Conics, parametric and non-parametric forms; Curves and Splines; Surfaces and NURBS, 3-D modelling (8 Lectures)
Hidden Surfaces: Depth comparison, Z-buffer algorithm, Back face detection, BSP tree method, the Printer’s algorithm, scan-line algorithm; Hidden line elimination, wire frame methods (6 Lectures)
Color and Shading Models:Phong's shading model, Gouraud shading, Shadows and background, Color models, Photo-realistic rendering, Radiosity (5 Lectures)
Animation: Functions, pipeline, sample programs for drawing 2-D, 3-D objects; event handling and view manipulation (5 Lectures)

Text Book
1. D. HEARN, P. BAKER (2002), Computer Graphics, Pearson Education India, 2002, 2nd Edition.

Reference Books
1. J. F. HUGHES, A. VAN DAM, M. McGUIRE, D.F. SKLAR, J. D. FOLEY, S. K. FEINER, K. AKELEY (2014), Computer Graphics: Principles and Practices, Addison Wesley, 3rd Edition.
2. D. F. ROGERS, J. A. ADAMS (1989), Mathematical Elements for Computer Graphics, McGraw Hill, 2nd Edition.
3. Z. XIANG, R. PLASTOCK (2015), Schaum’s Outline of Computer Graphics, McGraw Hill Education, 2nd Edition.
4. J. KESSENICH, G. SELLERS, D. SHREINER (2017), OpenGL Programming Guide, Pearson Education, 9th Edition.

Self Learning Material
NPTEL Computer Science and Engineering - Computer Graphics: https://nptel.ac.in/courses/106106090/.

Title	**Social Networks**	Course No.	CSL4XX0
Department	Computer Science and Engineering	Structure (L-T-P [C])	3–0–0 [3]
Offered for	B. Tech	Type	Elective
Prerequisite	None	Antirequisite	Social Network Analysis

Objectives
Provide introduction to social network analysis, its mathematical foundation and application.

Learning Outcomes
The students are expected to:
1. Gain the ability to understand the applications related to social networks
2. Write programs with social network datasets and Formulate real-world problems with any relational data set resembling social networks

Contents
Introduction*:* Graphs, Social Networks, Network Types, Network Data Sets, Gephi for Network Analysis (5 Lectures)
Network Properties*:* Network Measures, Strong and Weak Ties, Homophily, Structural Balance, Components (4 Lectures)
Network Models*:* Random Networks, Scale Free Networks, The Barabási-Albert Model, Erdos-Renyi Model (5 Lectures)
Structural Analysis of Networks using Python*:* Python for Network Analysis, Empirical Studies, Structural
Properties, Generate Synthetic Networks, Working with signed networks (5 Lectures)
Social Network Applications*:* Information Cascades, Small-World Phenomenon, Epidemics, Community Detection, Link Prediction, Page Rank (14 Lectures)
Evolving Network and Temporal Networks*:* Network evolution, working with Temporal Network Data (5 Lectures)
Multiplex and Multi-layer network (1 Lecture)
Network Analysis in Other Fields:Network Analysis in Biology, Sports, Transports (3 Lectures)

Textbooks

1. D. EASLEY, J. KLEINBERG (2010), Networks, Crowds, and Markets: Reasoning About a Highly Connected World, Cambridge University Press. (http://www.cs.cornell.edu/home/kleinber/networks-book/)
2. A. BARABASI (2016), Network Science, Cambridge University Press. (http://barabasi.com/networksciencebook/)
3. M. NEWMAN (2018), Networks, Oxford University Press, 2nd Edition. (https://global.oup.com/academic/product/networks-9780198805090?cc=us&lang=en&#/)

Reference Books

1. C. GROS (2015), Complex and Adaptive Dynamical Systems, Springer, Springer, 4th Edition.
2. E. ESTRADA (2011), The Structure of Complex Networks Theory and Applications, Oxford University Press.
3. W. de NOOY, A. MRVAR, V. BATAGELJ (2018), Exploratory Social Network Analysis with Pajek, Cambridge University Press, 3rd Edition.

Self Learning Material
1. https://www.barabasilab.com/course
2. https://nptel.ac.in/courses/106106169/#

Course Title	**Software Maintenance**	Course No.	CS4xxx
Department	Computer Science & Engineering	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B.Tech	Type	Elective
Prerequisite	Software Engineering	Antirequisite	None

Objectives

1. To explain theories, models, tools, and processes for the maintenance and evolution of large software systems and cloud applications
2. To familiarize with practices, techniques and processes that can help team build high quality software

Learning Outcomes

1. Understand the difficulties of developing code in a change context as opposed to new development
2. Identify the principal issues associated with software evolution and explain their impact on the software lifecycle
3. Describe techniques, coding idioms and other mechanisms for implementing designs that are more maintainable

Contents

Introduction: Basic concepts, difference between new development and maintenance, categorizing software change and limitations, maintenance frameworks (4 lectures)

Understanding Software & Evolution: Program comprehension models: top-down, bottom-up, opportunistic; Variants of software maintenance, relations to software engineering life-cycle, techniques of software evolution and testing, quantitative analyses, empirical analyses, qualitative analyses of software, evolution dynamics (10 Lectures)

Software Maintenance: Maintenance processes: quick-fix model, boehm's model, osborne's model, iterative enhancement model, process maturity, Search-based Software Testing, Summarization Techniques for Code, Changes, and Testing, Change impact analysis (7 Lectures)

Software Reuse: Reverse engineering levels & techniques: forward, restructuring, refactoring, reengineering, software reuse and evolution, the laws of software evolution, reuse techniques and design for reuse, libraries vs. application frameworks - Software product lines (7 Lectures)

Laboratory

Assignments/Projects on program comprehension and reverse engineering: program slicer, static analyser, dynamic analyzer, dataflow analyser, cross-reference, dependency analyser, transformation tools; Testing tools: simulators, test case generator, test path generators; source code, control system.

Text Books

1. Grubb, P., & Takang, A. A. (2003). Software maintenance: concepts and practice. World Scientific.
2. Galin, D. (2018). Software quality: concepts and practice. John Wiley & Sons..

Reference Book

Pigoski, T. M. (1996). Practical software maintenance: best practices for managing your software investment. Wiley Publishing.

Online Courses

Reverse Engineering Essentials: Udemy,
https://www.udemy.com/course/reverse-engineering-essentials/

Title	**Visual Computing Lab**	Number	CSE/EE 4XX
Department		L-T-P [C]	0–0–4 [2]
Offered for	BTech.	Type	Specialization Core
Prerequisite	None	Antirequisite	None

Objectives
The Instructor will introduce students to the importance of human centered visual computing.

Learning Outcomes
The students are expected to have the ability to design applications that explicitly consider human perception.

Contents
Sensing Devices: Concept of dynamic range, HDR displays, Event Based Camera, Time of Flight Camera, Coded Apertures, LIDAR, 3D Reconstruction, Kinect Scanner, Laser Scanner (4 weeks)
Mini-project in Recognition (2 weeks) Mini-project in Navigation (2 weeks) Mini-project in Action (2 weeks)
Mini-project in Mixed Reality (2 weeks) Mini-project in Trustable Decision (2 weeks)

Textbooks
Szeliski, R. (2010). *Computer Vision: Algorithms and Applications*. Springer-Verlag New York Inc. Available Online.
Reinhard, E., Heidrich, W., Debevec, P., Pattanaik, S., Ward, G., &Myszkowski, K. (2010). *High Dynamic Range Imaging: Acquisition, Display, and Image-based Lighting*. 2nd Edition. Morgan Kaufmann.

Self Learning Material
Prof.Ioannis (Yannis) Gkioulekas, Computational Photography, Carnegie Mellon University,http://graphics.cs.cmu.edu/courses/15-463/
Prof. Silvio Savarese, Computer Vision, From 3D Reconstruction to Recognition, Stanford University,http://web.stanford.edu/class/cs231a/

Preparatory Course Material:
Prof. Gilbert Strang, Linear Algebra and Learning from Data, Massachusetts Institute of Technology,https://math.mit.edu/~gs/learningfromdata/

700 - Level Courses

• Preliminary Courses (CSE and AI&DS)

Title	**Bridge course on DSA**	Number	**CS7xx**
Department	Computer Science	L-T-P [C]	0–0–4 [2]
Offered for	B.Tech, M.Tech. and Ph. D.	Type	Core
Prerequisite		Antirequisite	

Objectives
1. To revise the topics covered in Undergrad Data structure, and Algorithm courses.
2. To implement introductory data structures and algorithms in C/C++/Python.

Learning Outcomes
1. Learn basic data structure and algorithmic techniques
2. Implementation of various data structure and algorithms to solve real life problems

Contents
Data Structures [3 days]: Stack, Queue, Linked Lists, Heaps (min and max) , Binary Search Tree.

Introduction to Algorithms [3 days]: Asymptotic notations, Recurrence relations, Sorting: quick, radix, selection and heap sort, Searching.

Graph Algorithms [3 days]: DFS in directed and undirected graphs, BFS and connected components, Connectivity in directed graphs, Dijkstra's algorithm, topological sorting

Greedy and Divide & Conquer Algorithms [3 days]: Prim's, Kruskal's, interval scheduling, Mergesort, Masters theorem.

Misc. [3 days]: Binary Trees, AVL trees, Dynamic programming - matrix multiplication.

Textbooks

1. Jon Kleinberg, Eva Tardos (2005), Algorithm Design, Pearson Education, 1st Edition.
2. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, Introduction to Algorithms.

Self Learning Material
1. https://nptel.ac.in/courses/106/102/106102064/

Title	**Data Structures and Practices**	Number	CSP7XX0
Department	Computer Science and Engineering	L-T-P [C]	0–0–2 [1]
Offered for	M.Tech. 1st Year	Type	Compulsory
Prerequisite	Computer Programming	Antirequisite	None

Objectives

The Instructor will:

1. Explain various data structures and provide details to implement and use them in different algorithms

Learning Outcomes

Students are expected to have the ability to:

1. Write, debug and rectify the programs using different data structures
2. Expertise in transforming coding skills into algorithm design and implementation

Contents

Laboratory Experiments

Exercises based on

Abstract Data Types: Arrays, link-list/list, hash tables, dictionaries, structures, *stack, queues* (4 labs)
Data Structures: Heap, Sets, Sparse matrix, Binary Search Tree, B-Tree/ B+ Tree, Graph (4 labs)
Algorithm implementation: Quick or Merge sort, Breadth or Depth first search or Dijkstra's Shortest Path First algorithm, Dynamic programing (6 labs)

Textbook

1. Weiss, M. A. (2007), Data Structures and Algorithm Analysis in C++, Addison-Wesley.
2. Lipschutz, S. (2017), Data Structures with C, McGraw Hill Education.
3. Cormen, T. H., Leiserson, C. E., Rivest, R. L. and Stein, C., (2009), Introduction to Algorithms, MIT Press

Online Course Material

1. Department of Computer Science and Engineering, IIT Delhi, http://www.nptelvideos.in/2012/11/data-structures-and-algorithms.html

700 - Level Courses

• Core Courses (CSE and AI&DS)

Title	**Advanced Data Structures & Algorithms**	Number	**CS7xx**
Department	Computer Science	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech, M.Tech. CSE and M.Tech. AI, Ph.D.	Type	Core
Prerequisite	Algorithm Design and Analysis, Data Structures, Maths for Computing	Antirequisite	None

Objectives

1. The objective of the course is to introduce several advanced algorithmic techniques and data structures.
2. Implementation of various advanced data structure and algorithms to solve real life problems

Learning Outcomes

Students are expected to have the ability to:

1. Learn advanced data structure techniques
2. Learn a new set of techniques to cope with NP-hard problems.
3. Identify novel and significant open research questions in the field.
4. Implementation of various advanced data structure and algorithms to solve real life problems
5. Understand the intractability of problems.

Contents

Data Structures and Algorithms Techniques (Fractals I and II)

Advanced Data Structures [10 lectures]: Red-black trees, Interval trees, Dynamic order statistics, Binomial heap, Fibonacci Heap [5 lectures], Tries, Splay Trees, B-Trees, Hashing - disjoint set, union find [5 lectures].

Network Flows [5 lectures]: Maximum Flow problem, Ford-Fulkerson algorithm, Max flow min cut theorem, augmenting paths, applications to bipartite matching problem and disjoint paths in directed and undirected graphs.

Amortized Analysis [3 lectures]: Aggregate Analysis, Accounting method, Potential method.

Dynamic Programming [6 lectures]: Knapsack, Independent Set of Trees, Interval Scheduling, Set Cover, Permutation Problems, Partition Problems, String matching: naive method, Knuth-Morris-Pratt algorithm.

NP-completeness and Reductions [8 lectures]: Complexity classes P, NP, co-NP, NP-complete, NP-Hard, Example reductions between problems.

Fractal III: Advanced Algorithmic Techniques

Advanced Algorithmic Techniques [10 lectures]: Approximation Algorithms: Greedy Algorithm – Load Balancing, Set Cover [2 Lectures]; The Pricing Method: Vertex Cover, Knapsack [3 Lectures], Randomized Algorithms [5 Lectures].

Textbooks

1. Jon Kleinberg, Eva Tardos (2005), Algorithm Design, Pearson Education, 1st Edition.
2. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, Introduction to Algorithms.

Self Learning Material

1. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-046j-introduction-to-algorithms-sma-5503-fall-2005/video-lectures/
2. https://nptel.ac.in/courses/106/101/106101060/
3. https://nptel.ac.in/courses/106/104/106104019/

Title	**Algorithms for Big Data**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B. Tech., M.Tech., PhD	Type	Compulsory
Prerequisite	None	Antirequisite	None

Objectives
The Instructor will:
1. Introduce some algorithmic techniques developed for handling large amounts of data.
2. Emphasize both theoretical as well as practical aspects of such algorithms.

Learning Outcomes
Students are expected to have the ability to:
1. Design and Analyzing existing algorithms as well as design novel algorithms pertaining to big data.

Contents:
Introduction: Randomized algorithms, Universal hash family, Probabilistic algorithm analysis, Approximation algorithms,ε-Approximation schemes, Sublinear time complexity, Sublinear Algorithms. (7 lectures)
Property Testing: Testing list's sortedness or monotonicity, Distribution testing (5 lectures)
Testing properties of bounded degree graphs, Dense graphs and General graphs. (6 lectures)
Sketching and Streaming: Extremely Small-Space Data Structures, CountMin Sketch, Count Sketch (5 lectures)
Linear Sketching, AMS Sketch, Turnstile Streaming, Graph Sketching, Graph Connectivity (5 lectures)
MapReduce: MapReduce Algorithms in Constrains Settings such as small memory, few machines, few rounds, and small total work, Efficient Parallel Algorithms (7 lectures)
External memory and cache-obliviousness: Minimizing I/O for large datasets, Algorithms and data structures such as B-trees, Buffer trees, Multiway merge sort (7 lectures)

Self Learning Material
1. Department of Computer Science, Harvard University, Algorithms for Big Data
2. https://www.sketchingbigdata.org/
3. "Introduction to Property Testing" (link) by Oded Goldreich
4. http://grigory.us/big-data.html

Title	**Artificial Intelligence (700)**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B. Tech., M.Tech., Ph.D.	Type	Compulsory (AI)
Prerequisite	Data Structures and Algorithms	Antirequisite	Artificial Intelligence (300) - CSL 3XX

Objectives
The Instructor will:
1. Cover various paradigms that come under the broad umbrella of AI.

Learning Outcomes
The students are expected to have the ability to:
1. Develop an understanding of where and how AI can be used.

Contents
Introduction: Uninformed search strategies**,** Greedy best-first search, And-Or search, Uniform cost search, A* search, Memory-bounded heuristic search, Local and evolutionary searches (9 Lectures)
Constraint Satisfaction Problems: Backtracking search for CSPs, Local search for CSPs (3 Lectures)
Adversarial Search: Optimal Decision in Games, The minimax algorithm, Alpha-Beta pruning, Expectimax search (4 Lectures)
Knowledge and Reasoning: Propositional Logic, Reasoning Patterns in propositional logic; First order logic: syntax, semantics, Inference in First order logic, unification and lifting, backward chaining, resolution (9 Lectures)
Planning: Situation Calculus, Deductive planning, STRIPES, sub-goal, Partial order planner (3 Lectures)
Bayesian Network, Causality, and Uncertain Reasoning: Probabilistic models, directed and undirected models, inferencing, causality, *Introduction to Probabilistic reasoning* (6 lectures)
Reinforcement Learning: MDP, Policy, Q-value, Passive RL, Active RL, Policy Search (8 Lectures)

Textbook
1. Russel,S., and Norvig,P., (2015), *Artificial Intelligence: A Modern Approach*, 3rd Edition, Prentice Hall

Reference Books
1. Research literature

Self Learning Material
1. Department of Computer Science, University of California, Berkeley, http://www.youtube.com/playlist?list=PLD52D2B739E4D1C5F
2. NPTEL: Artificial Intelligence, https://nptel.ac.in/courses/106105077/

Title	**Computer Architecture (700)**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	M.Tech, PhD	Type	Compulsory
Prerequisite	none	Antirequisite	Computer Architecture (300)

Objectives

The Instructor will:

1. Provide background to understand various components of a modern computer system, its interconnections, and performance issue

Learning Outcomes

The students are expected to have the ability to:

1. Explain the working principles of various components of modern computer systems
2. Compare between systems using benchmark data
3. Write, execute and debug parallel programs on GPU

Contents

CSL7XX0 Multi System Architecture 1-0-0 [1]

Introduction: Defining Computer Architecture, Flynn's Classification of Computers, Metrics for Performance Measurement. (4 lectures)

Memory Hierarchy: Introduction, Advanced Optimizations of Cache Performance, Memory Technology and Optimizations, Virtual Memory and Virtual Machines, The Design of Memory Hierarchies, Introduction to Pin Instrumentation and Cache grind, Case Study: Intel Core i7 (10 lectures)

CSL7XX0 Multicore Processing 1-0-0 [1]

Instruction-Level Parallelism: Instruction-level Parallelism: Concepts and Challenges, Basic Compiler Techniques for Exposing ILP, Reducing Branch Costs with Advanced Branch Prediction, Dynamic Scheduling, Superscalar, Limitations of ILP, Case Study: Dynamic Scheduling in Intel Core i7. (9 lectures)

Multicore Processor: Introduction, CPU Interconnections, Network on Chip (NoC), Routing Protocols, Quality of Service on NoC. (5 lectures)

CSL7XX0 Fundamentals of Parallel Programming 1-0-0 [1]

Data Level Parallelism: Introduction, Vector Architecture, SIMD Instruction Set Extensions for Multimedia, Graphics Processing Units, GPU Memory Hierarchy, Detecting and Enhancing Loop- Level Parallelism, CUDA Programming, Case Study: Nvidia Maxwell. (14 lectures)

Textbook

1. Hennessy,J.L. and Patterson,D.A., (2012), *Computer Architecture: A Quantitative Approach*, 5th Edition, Morgan Kaufmann Publishers
2. Shen,J.P. and Lipasti,M.H., (2005), *Modern Processor Design: Fundamentals of Superscalar Processors*, McGraw-Hill Publishers

Self-Learning Material

1. *CUDA:* https://developer.nvidia.com/cuda-zone
2. *OpenMP:* https://www.openmp.org/

Preparatory Course Material

1. Department of Computer Science and Engineering, Indian Institute of Technology Madras, https://nptel.ac.in/courses/106106134/

Title	**Deep Learning (700)**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	M.Tech. 1st Year, Ph.D. 1st Year	Type	Compulsory (AI)
Prerequisite	Machine Learning	Antirequisite	Deep Learning (400) - CSL4xx

Objectives

1. Provide technical details about various recent algorithms and software platforms related to Machine Learning with specific focus on Deep Learning.

Learning Outcomes

Students are expected to have the ability to:

1. Design and program efficient algorithms related to recent machine learning techniques, train models, conduct experiments, and develop real-world DL-based applications and products

Contents

Fractal 1: Foundations of Deep Learning

Deep Networks: CNN, RNN, LSTM, Attention layers, Applications (8 lectures)

Techniques to improve deep networks: DNN Optimization, Regularization, AutoML (6 lectures)

Fractal 2: Representation Learning

Representation Learning: Unsupervised pre-training, transfer learning, and domain adaptation, distributed representation, discovering underlying causes (8 lectures)

Auto-DL: Neural architecture search, network compression, graph neural networks (6 lectures)

Fractal 3: Generative Models

Probabilistic Generative Models: DBN, RBM (3 lectures)

Deep Generative Models: Encoder-Decoder, Variational Autoencoder, Generative Adversarial Network (GAN), Deep Convolutional GAN, Variants and Applications of GANs (11 lectures)

Text Book

1. Goodfellow,I., Bengio.,Y., and Courville,A., (2016), Deep Learning, The MIT Press .

Reference Book

1. Charniak, E. (2019), Introduction to deep learning, The MIT Press.
2. Research literature.

Self Learning Material

1. https://www.deeplearningbook.org/

Title	**Dependable AI (700)**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech, M.Tech., PhD	Type	Elective
Prerequisite	Machine Learning, Artificial Intelligence	Antirequisite	Dependable AI (400) - CSL 4xx

Objectives

The Instructor will:

1. Provide characteristic details of AI and machine learning systems to make them dependable, such as explainability, interpretability, safety etc.

Learning Outcomes

The students are expected to have the ability to:

1. Assess the dependability of AI systems.
2. Develop explainable, robust, and safe AI models.

Contents

Introduction: Overview, Motivation, Challenges – medical and surveillance (3 Lectures)
Explainable AI: Accuracy-explainabilityTradeoff, Interpretability Problem, Predictability, Transparency, Traceability, Causality, Reasoning, Attention and Saliency (10 Lectures)
Interpretable AI: Prediction Consistency, Application Level Evaluation, Human Level Evaluation, Function Level Evaluation (5 Lectures)
Adversarial Robustness: Adversarial Attacks and Defences (8 Lectures)
Trustworthy AI: Integrity, Reproducibility, Accountability (2 Lectures)
Bias-free AI: Accessibility, Fair, Data Agnostics Design, Disentanglement (4 Lectures)
Privacy Preserving AI:Federated Learning, Differential Privacy and Encrypted Computation (6 Lectures)
Verified AI:Environment and Specification Modeling, Design with Formal Inductive Synthesis, Evaluation Platforms for AI Safety (4 Lectures)

Textbook

1. Pearl, J., (2018), *The Book of Why: The New Science of Cause and Effect*, Basic Books.

Reference Book

1. Bostrom, N., (2014), *The Ethics of Artificial Intelligence. The Cambridge handbook of artificial intelligence*, Cambridge University Press.

Preparatory Course Material

1. Proceedings of IJCAI: Workshop on Explainable Artificial Intelligence (XAI).

Title	**DL-Ops**	Number	**CS7xx**
Department	Computer Science	L-T-P [C]	0–0–2 [1]
Offered for	B.Tech. (CSE and AI), M.Tech. (CSE and AI), Ph.D.	Type	Core
Prerequisite	Machine Learning or PRML or IML	Antirequisite	None

Objectives
- To cover various techniques of Deep Learning system design.
- To cover topics around building a DL system that is deployable, reliable, and scalable

Learning Outcomes
Students will gain the ability to:
1. Develop an understanding to analyze tradeoffs for designing production systems with DL-components, analyzing various qualities beyond accuracy such as operation cost, latency, and updateability
2. Understanding of latest trends in systems designs to better support the next generation of DL applications, and applications of DL to optimize the architecture and the performance of systems.
3. Develop an understanding to design production-quality, fault tolerance, scalable DL system

Contents
- Deep Learning Frameworks, Hardware Requirements, Containers, Versioning, Provenance, and Reproducibility, DL Compilers (4 labs)
- Large Data Requirement (Storage, Lineage), Distributed and Scalable Deep Learning, Dynamic Networks, Parallel Optimization, Mixed Precision Training (4 labs)
- Deployment, Monitoring, Updating, and Maintenance (2 labs)
- Hyperparameter Tuning, Faster Serving, DL Systems in Production and Client Level Inference (e.g NVIDIA Triton) (2 labs)
- Advanced Topics (2 labs)

Textbooks
- None - online materials and sysML level papers

Self Learning Material
- https://stanford-cs329s.github.io/
- https://ucbrise.github.io/cs294-ai-sys-sp19/
- https://ckaestne.github.io/seai/
- https://www.coursera.org/specializations/machine-learning-engineering-for-production-mlops#courses

Title	**Machine Learning (700)**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	M.Tech. (CSE, AI, DCS), PhD	Type	Compulsory
Prerequisite	Introduction to Computer Sc., Probability, Statistics and Stochastic Processes, Linear Algebra	Antirequisite	IML, PRML

Objectives

1. To understand various key paradigms for machine learning approaches
2. To familiarize with the mathematical and statistical techniques used in machine learning.
3. To understand and differentiate among various machine learning techniques.

Learning Outcomes

The students are expected to have the ability to:

1. To formulate a machine learning problem
2. Select an appropriate pattern analysis tool for analyzing data in a given feature space.
3. Apply pattern recognition and machine learning techniques such as classification and feature selection to practical applications and detect patterns in the data.

Contents

Fractal I: Supervised Learning

Introduction: Definitions, Datasets for Machine Learning, Different Paradigms of Machine Learning, Data Normalization, Hypothesis Evaluation, VC-Dimensions and Distribution, Bias-Variance Tradeoff, Regression (Linear) (7 Lectures)
Bayes Decision Theory: Bayes decision rule, Minimum error rate classification, Normal density and discriminant functions (5 Lectures)
Parameter Estimation: Maximum Likelihood and Bayesian Parameter Estimation (2 Lectures)

Fractal II: Unsupervised Learning

Discriminative Methods: Distance-based methods, Linear Discriminant Functions, Decision Tree, Random Decision Forest and Boosting (6 Lectures)
Feature Selection and Dimensionality Reduction: PCA, LDA, ICA, SFFS, SBFS (4 Lectures)
Clustering: k-means clustering, Gaussian Mixture Modeling, EM-algorithm (4 Lectures)

Fractal III: Kernels and Neural Networks

Kernel Machines: Kernel Tricks, SVMs (primal and dual forms), K-SVR, K-PCA (6 Lectures)
Artificial Neural Networks: MLP, Backprop, and RBF-Net (4 Lectures)
Foundations of Deep Learning: DNN, CNN, Autoencoders (4 lectures)

Text Book

1. Shalev-Shwartz,S., Ben-David,S., (2014), Understanding Machine Learning: From Theory to Algorithms, Cambridge University Press
2. R. O. Duda, P. E. Hart, D. G. Stork (2000), Pattern Classification, Wiley-Blackwell, 2nd Edition.

Reference Book

1. Mitchell Tom (1997). Machine Learning, Tata McGraw-Hill
2. C. M. BISHOP (2006), Pattern Recognition and Machine Learning, Springer-Verlag New York, 1st Edition.

Self-Learning Material

1. Department of Computer Science, Stanford University, https://see.stanford.edu/Course/CS229

Title	**ML-Ops**	Number	**CSP7xx**
Department	Computer Science	L-T-P [C]	0–0–2 [1]
Offered for	B.Tech., M.Tech., Ph.D.	Type	Core
Prerequisite	Machine Learning or PRML or IML	Antirequisite	None

Objectives

- To cover topics of ML system design, including applications and products with machine learning algorithms.

Learning Outcomes

Students will gain the ability to:

1. Learning to design ML Systems to solve practical problems.
2. Learning how an ML system works in production and insights about challenges
3. Identifying systems faults and applying strategies to identify root causes in ML systems.
4. Picking the right framework and compute infrastructure and trade-off space.
5. Troubleshooting training and ensuring the reproducibility of results.

Contents

- Machine Learning Systems: Concepts and Stages (data collection to model development), Challenges, and Solutions, DevOps to MLOps - Framework and Software Stack (2 labs)
- ML Data Structure and Data Processing (2 labs)
- Machine Learning Accelerators (ML Compilers), Virtual Environments, Git, Docker, Containers, Kubernetes (2 lab)
- Experiment Tracking, Reproducibility, and Reusability (2 labs)
- Quantized and Low-precision Machine Learning (2 labs)
- Deployment: Platforms and Infrastructure (2 labs)
- Machine Learning System Versioning, Tracking, Testing and Debugging (2 labs)

Textbooks

- Online materials and sysML level papers

Self Learning Material

- https://stanford-cs329s.github.io/
- https://ucbrise.github.io/cs294-ai-sys-sp19/
- https://ckaestne.github.io/seai/
- https://www.coursera.org/specializations/machine-learning-engineering-for-production-mlops#courses

Title	**Software and Data Engineering**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	M.Tech., PhD	Type	Compulsory
Prerequisite	Data Structures and Algorithms	Antirequisite	None

Objectives
The Instructor will:
1. Discuss techniques to manage a large amount of data
2. Provide mechanisms to design and develop data-intensive computing systems

Learning Outcomes
The students are expected to have the ability to:
1. Design complex end-to-end data pipeline for data processing
2. Critically identify and use the tools for data handling and management
3. Use modern software technologies to design and develop data analytical systems

Contents
CSL7XX0 Cloud Computing and Virtualization 1-0-0 [1]
Basics of complex software design: Concept of modular software, microservices, communication, 4+1 architectural views and patterns (5 lectures)
Cloud Computing: Architecture of cluster computing, design of data centers, open data center platforms, fault-tolerant system design (5 lectures)
Virtualization: Type-1 and Type-2 virtualization, virtual machine, containers, dockers (4 lectures)
CSL7XX0 Data Management 1-0-0 [1]
Data Management: Structured data, relational database management, unstructured data, semi-structured data, Nosql database management (mongodb), column database, graph database, XML, JSON, HDFS, Handling drift in data, sensor data reliability at software and algorithmic level, sensor data analysis techniques (14 lectures)
CSL7XX0 Data Intensive Processing Systems 1-0-0 [1]
Data Intensive Processing Systems: Architecture of large scale data processing systems, Hadoop, Apache Spark, Storm, parallel data processing concepts such as map-reduce, directed acyclic graph, resilient distributed datasets, dynamic resource allocation, partial & shared computation, storage architecture (14 lectures)

Textbook
1. Bass L., Clements P., Kazman R., (2012), *Software Architecture in Practice*, 3rd edition, Addison-Wesley Professional
2. Martin K., (2017), *Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems*, 1st Edition, O'Reilly Media

Self Learning Material
1. Tylor,R.N., Medvidovic,N. and Dashofy,E.M., (2014), *Software Architecture Foundation: Theory and Practice*, Wiley

Preparatory Course Material
1. IEEE Transactions on Knowledge and Data Engineering
2. International Conference on Data Engineering

700 - Level Courses

• Elective Courses (CSE and AI&DS)

Title	**Advanced Artificial Intelligence (700)**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B. Tech., M.Tech., Ph.D.	Type	Elective
Prerequisite	Data Structures and Algorithms	Antirequisite	None

Objectives

The Instructor will:

1. Cover modern paradigms of AI that go beyond traditional learning.

Learning Outcomes

The students are expected to have the ability to:

1. Develop an understanding of modern concepts in AI and where they can be used.
2. Design, implement and apply novel AI techniques based on emerging real-world requirements.

Contents

Decision Making: Utility theory, utility functions, decision networks, sequential decision problems, Partially Observable MDPs, Game Theory (14 Lectures)
Probabilistic Reasoning over time: Hidden Markov Models, Kalman Filters, Time Series Analysis (10 Lectures)
Knowledge Representation: Information extraction, representation techniques, foundations of Ontology, Ontological engineering, Situation Calculus, semantic networks, description logic (8 Lectures)
Planning: Planning with state space search, Partial-Order Planning, Planning Graphs, Planning with Propositional Logic, hierarchical task network planning, non-deterministic domains, conditional planning, continuous planning (6 Lectures)
Embodied AI: Introduction to visual navigation (synthetic and real environment), Embodied question answering in Synthetic Interactive and indoor environment, Introduction to AI Habitat simulation platform. (4 Lectures)

Textbook

1. Russel,S., and Norvig,P., (2015), *Artificial Intelligence: A Modern Approach*, 3rd Edition, Prentice Hall

Reference Books

1. Yang, Q. (1997), *Intelligent Planning: A decomposition and abstraction based approach,* Springer Verlag, Berlin Heidelberg.

Course Title	**Advanced Biometrics**	Course No.	**CSL7430**
Department	Computer Science and Engineering, AI&DS	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B.Tech, MTech, PhD	Type	Elective
Prerequisite	Pattern Recognition and Machine Learning / Deep Learning / Machine Learning	Antirequisite	None

Objectives

1. To familiarize the students with types of biometrics including physical and behavioural modalities, understanding biometric strengths, weaknesses and limitations, and biometric standards.
2. Describe a few techniques for designing biometric systems

Learning Outcomes

The students are expected to have the ability to:

1. Develop new biometric systems for real-world applications, and become familiar with various stages of biometric system development.
2. Develop an understanding of future direction and emerging technologies in biometrics.

Contents

Introduction:Biometric system design, Genesis of biometrics: human body properties, and biometric data acquisition, System architecture, Performance Evaluation: Verification, Identification (4 Lectures)
Traditional and Emerging Biometric Modalities: Image/signal processing, Fingerprint matching, face recognition, iris modelling, signature authentication, biometric pattern recognition, multi-modal biometrics (23 Lectures)
Biometric Security: Encryption, cancelable biometrics and fuzzy vault (6 Lectures)
Biometric Devices: Security-Target design, Reliability design, Industry standards (6 Lectures)
Biomedical applications of Biometrics (3 Lectures)

Text Book

1. A.K. JAIN, A. ROSS, K. NANDAKUMAR (2011), Introduction to Biometrics, Springer.

Reference Books

1. A.K. JAIN, A. ROSS, K. NANDAKUMAR, (Eds.), Handbook of Biometrics, Springer, 2008.
2. S.Z. LI, A.K. JAIN, (Eds.), Handbook of Face Recognition, Springer, 2011, 2nd Edition.
3. D. MALTONI, D. MAIO, A. JAIN, S. PRABHAKAR, (Eds.), Handbook of Fingerprint Recognition, Springer, 2009, 2nd Edition.
4. M.J. BURGE, K.W. BOWYER, (Eds.), Handbook of Iris Recognition, Springer, 2016, 2nd Edition.
5. On-line resources will be provided.

Title	**Advanced Computer Graphics**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech, M.Tech., PhD	Type	Elective
Prerequisite	Computer Graphics	Antirequisite	

Objectives

The Instructor will:

1. Discusses fundamentals of 2D and 3D object modeling and rendering
2. Explain the Hardware and software related issues of Computer Graphics

Learning Outcomes

The students are expected to have the ability to:

1. Understand fundamentals of graphics used in various real life applications
2. Identify the performance characteristics of graphics algorithms
3. Employ algorithm to model engineering problems, when appropriate

Contents

Introduction: Review of 2-D, and 3D Geometry, Viewing and Clipping (5 lectures)

Curves and Fractals: Parametric Cubic curves: B-spline, Bezier, Hermite, Surfaces, Fractals and its applications (9 lectures)

Solid Modeling: Representation of Solids, Sweep and Boundary Representation, Constructive Solid Geometry (8 lectures)

Illumination and Shading: Surface detail, shadows and Transparency, Inter object Reflections, Illumination Models, Extended Light Sources, Ray Tracing, Radiosity (6 lectures)

Image Based Rendering: Image synthesis, Geometry based, Plenoptic Function, Panorama, Lumigraph, Rendering Virtual Reality (8 lectures)

Animation: Introduction, morphing, character animation and facial animation (3 lectures)

Graphics Hardware: Special-purpose computer graphics processors and accelerators (3 lectures)

Textbook

1. Shirley,M., (2016), *Fundamentals of Computer Graphics*, 4th Edition, CRC Press
2. vanDam,F. and Hughes,F., (2013), *Computer Graphics: Principles and Practice*, 3rd Edition, Addision Wesley

Reference Books

1. Mukundan,R., (2012), *Advanced Methods in Computer Graphics: With Examples in OpenGL*, Springer
2. Ruben H., (2017), *Computer Graphics: Principles and Practice*, Larsen and Keller Education

Self Learning Material

1. Computer Graphics, NPTEL Video Lectures, https://nptel.ac.in/courses/106106090/

Course Title	**Advanced Machine Learning**	Course No.	**CSL7xx**
Department	Computer Science and Engineering	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B.Tech, MTech, PhD	Type	Elective
Prerequisite	PRML/Introduction to ML, AI	Antirequisite	None

Objectives

This is an advanced course on Machine Learning. It is intended for senior undergraduate and graduate students who already have a background in PR or ML. The course is designed as a set of special topics that will be covered through research papers and books. The course will focus on algorithms and models along with providing a good perspective of different real world applications.

Learning Outcomes

The students are expected to have the ability to:

1. Understand the algorithms and functioning of advanced techniques and concepts such as deep learning, distance metric learning, and domain adaptation
2. Understand the advantages and limitations of the algorithms and their potential applications
3. Run several public domain machine learning toolboxes on real world databases such as MNIST and CIFAR10
4. Design experiments for evaluation and analyze the results to test the effectiveness of individual components of an algorithm

Contents

Kernel Machines: Kernel properties, Kernels for structure data and text, Multiple kernel learning, Generative models (3 Lectures)

Variants of Support Vector Machine: Hard and soft margin SVM, Online SVM, Distributed SVM (3 Lectures)

PAC Theory (6 Lectures)

Boosting: Adaboost, Gradient boosting (2 Lectures)

Structured Prediction and Graphical Models: Learning directed and undirected models, Sampling, MAP inference and prediction, variational inference, causality (14 Lectures)

Dictionary Learning: Fundamentals, Regularization, Supervised and unsupervised dictionary, learning, Transform learning (6 Lectures)

Deep Reinforcement Learning (8 Lectures)

Text Books

1. N. CRISTIANINI, J. S-TAYLOR (2000), An Introduction to Support Vector Machines and Other Kernel- based Learning Methods, Cambridge University Press, 1st Edition.
2. B. SCHOLKOPF, A. J. SMOLA (2001), Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond, The MIT Press, 2001, 1st Edition.
3. R. S. SUTTON, A. G. BARTO (2018), Reinforcement Learning: An Introduction, The MIT Press, 2nd Edition.
4. D. KOLLER, N. FRIEDMAN (2009), Probabilistic Graphical Models: Principles and Techniques, MIT Press.

Title	**Ad-Hoc Wireless Networks**	Number	CSL7XX0
Department	CSE, EE	L-T-P[C]	3–0–0 [3]
Offered for	B. Tech., M.Tech., PhD	Type	Elective
Prerequisite	Networks	Antirequisite	None

Objectives
The Instructor will:
1. Introduce the mathematical models and network protocol designs in wireless Ad-hoc networks
2. Provide a systematic exposition of network protocols and their cross-layer interactions
3. To provide more advanced in-depth networking knowledge. Upon completion of this course, students will be able to apply the knowledge in their networking research. A broad perspective on the active research areas in wireless Ad-hoc networks

Learning Outcomes
The students are expected to have the ability to:
1. Demonstrate advanced knowledge of networking and wireless networking in particular
2. Compare different solutions for communications at each network layer
3. Demonstrate knowledge of protocols used in wireless communications

Contents
Basics of wireless networks and mobile computing: Ad hoc Networks: Introduction, Issues in Ad hoc wireless networks, Ad hoc wireless internet (3 lectures)
Media access control in ad hoc and sensor networks: MAC Protocols for Ad hoc Wireless Networks: Introduction, Issues in designing a MAC protocol for Ad hoc Wireless Networks, Design goals of a MAC protocol for Ad hoc Wireless Networks, Classification of MAC protocols, Contention based protocols with reservation mechanisms. Contention-based MAC protocols with Scheduling mechanism, MAC protocols that use directional antennas, Other MAC protocols, Network and transport layer issues for ad hoc and sensor networks (8 lectures)
Routing protocols for Ad hoc Wireless Networks: Introduction, Issues in designing a routing protocol for Ad hoc Wireless Networks, Classification of routing protocols, Table drive routing protocol, On-demand routing protocol, Hybrid routing protocol, Routing protocols with effective flooding mechanisms, Hierarchical routing protocols, Power aware routing protocols (8 lectures)
Transport layer protocols: Transport layer protocols for Ad hoc Wireless Networks: Introduction, Issues in designing a transport layer protocol for Ad hoc Wireless Networks, Design goals of a transport layer protocol for Ad hoc Wireless Networks, Classification of transport layer solutions, TCP over Ad hoc Wireless Networks, Other transport layer protocols for Ad hoc Wireless Networks (8 lectures)
Security issues for ad hoc networks: Security: Security in wireless Ad hoc Wireless Networks, Network security requirements, Issues & challenges in security provisioning, Network security attacks, Key management, Secure routing in Ad hoc Wireless Networks (6 lectures)
QoS for ad hoc Networks: Quality of service in Ad hoc Wireless Networks: Introduction, Issues and challenges in providing QoS in Ad hoc Wireless Networks, Classification of QoS solutions, MAC layer solutions, network layer solutions (3 lectures)
Advanced Topics: Software-defined network (SDN), Mesh networking, Energy issues and Sensor networks (6 lectures)

Laboratory Experiments Programming exercises using NS2/NS3, QualNet, Java and OmNet++

Textbook
1. Siva Ram Murthy, C., & Manoj, B. S. (2015). Ad hoc wireless networks: Architectures and protocols. *PHI Pearson Education*
2. Akyildiz, Ian F., and Xudong Wang(2015). *Wireless mesh networks*. Vol. 3. John Wiley & Sons

Reference Books
1. Basagni, S., Conti, M., Giordano, S., & Stojmenovic, I. (Eds.). (2015). Mobile ad hoc networking. John Wiley & Sons
2. Perkins, C. E. (2001). Ad hoc networking (Vol. 1). Reading: Addison-wesley

3. Toh, C. K. (2001). *Ad hoc mobile wireless networks: protocols and systems*. Pearson Education
4. Cheng, X., Huang, X., & Du, D. Z. (Eds.). (2013). *Ad hoc wireless networking* (Vol. 14). Springer Science & Business Media

Self Learning Material

1. Computer Networks - MIT OpenCourseWare https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-829-computer-networks-fall-2002/lecture-notes/
2. Mobile and Wireless Networks and Applications, Stanford University, https://web.stanford.edu/class/cs444n/

Course Title	**Approximation Algorithms**	Course No.	**CS7xxx**
Department	Computer Science and Engineering	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B.Tech, M. Tech, Ph.D.	Type	Bouquet/Elective
Prerequisite	Design and Analysis of Algorithms	Antirequisite	None

Objectives
The objective of the course is to introduce general-purpose techniques for large classes of intractable optimization problems.

Learning Outcomes
Students will be able to:
1. Identify novel and significant open research questions in the field.
2. Apply their knowledge in the design and implementation of viable solutions to those problems.

Contents
Combinatorial Techniques: Metric TSP, Euclidean TSP, FPTAS for knapsack, Greedy algorithms for makespan, PTAS for makespan, Local search: The Min degree spanning tree problem. (10 Lectures)
Linear Programming Relaxations: Linear programming: Vertex cover, Set cover via Dual fitting, Set cover via the Primal-Dual schema, Steiner forest, Facility location. (10 Lectures)
Cuts, Metrical Relaxations, and Embeddings: Basic embedding results and applications, Tree embeddings: Buy
at bulk network design and L1 embeddings, Min multicut and metrical relaxation for sparsest cut, Rounding for sparsest cut via L1 embeddings, Balanced cuts and their applications, Tree embeddings via random cuts. (10 Lectures)
Random Hyperplanes, Dimensionality Reduction, and Hashing: Solving semidefinite programs, Rounding SDPs using random projections: MAX-CUT, Dimensionality reduction via random projections, Approximate nearest neighbors, Locality sensitive hashing: Random projections and Min hashing. (10 Lectures)
Game Theory: Selfish algorithmic strategies, Cost sharing schemes using Primal-Dual algorithms. (2 Lectures)

Text book
V. VAZIRANI (2013), Approximation Algorithms, Springer-Verlag.

Reference Books
1. M.R. GAREY, D.S. JOHNSON (2016), Computers and Intractability: A Guide to the Theory of NP- Completeness, W. H. Freeman Company.
2. D. DU, K.KO, X. HU (2011), Design and Analysis of Approximation Algorithms, Springer.

Course Title	**Artificial Intelligence in Transportation**	Course No.	**CS7XX**
Department	Computer Science and Engineering	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B.Tech, MTech, PhD	Type	Elective
Pre-requisites	None	Antirequisite	None

Objectives
1. The course is intended to provide students with an understanding of the applications of AI in transportation systems.
2. The course would have a Computer Science flavor to transportation with relevant case studies.

Learning Outcomes
At the end of this course, students would be able to appreciate the tremendous AI-driven impact on transportation systems worldwide.

Contents
Traffic flow basics (3 Lectures)
Continuum models of traffic flow(4 Lectures)
Traffic modeling and control of freeways (3 Lectures)
Network-level traffic management (4 Lectures)
Control of traffic signals (4 Lectures)
ITS (Intelligent Transportation Systems) applications and case studies (4 Lectures)
Sensor technologies and data requirements of ITS(3 Lectures)
Overview of Autonomous Systems (4 Lectures)
Driverless Vehicles (3 Lectures)
Vehicular localization techniques (3 Lectures)
Future trends in transportation systems (3 Lectures)
Paper discussions (flipped mode) (4 Lectures)

Note:
Please note that the course material would not be taken from a single book or resource. It would be a combination of textbook material, research papers, and other sources.

Textbooks:
1. Alam, M., Ferreira, J., & Fonseca, J. (2016). Introduction to intelligent transportation systems. In Intelligent Transportation Systems (pp. 1-17). Springer, Cham.
2. Lipson, H., &Kurman, M. (2016). Driverless: Intelligent Cars and the road ahead. MIT Press.

Reference Books:
1. Chowdhury, M. A., & Sadek, A. W. (2003). Fundamentals of intelligent transportation systems planning. Artech House.
2. Recent relevant RFCs, Internet drafts, selected research papers from relevant venues: Mobicom, MobiSys, SIGCOMM, Infocom, IEEE TMC, ACM MC2R.

Self-learning Material:
1. https://www.edx.org/course/intro-to-traffic-flow-modeling-and-intelligent-tra
2. Relevant research papers.

Title	**Autonomous Systems**	Number	CS7xx
Department/IDRP	CSE	L-T-P [C]	3-0-0 [3]
Offered for	B. Tech, MTech, PhD (All Branches)	Type	Elective
Prerequisite	Basics of Linear Algebra, Probability and Statistics		

Objectives

1. Provide an understanding about autonomous/ semi-autonomous systems like autonomous mobile robots and cars.

Learning Outcomes

1. Understand and use the methodologies to design, model and implementation of autonomous systems for real time applications.

Contents:

Fractal I: Introduction to Autonomous Systems [14 Lectures]

- Introduction to Autonomous Systems, properties of autonomous systems, examples of autonomous systems, degree of autonomy **[3]**
- Abstract Agent Architectures, Environment and agent, reactive, utility-based and behavior based architectures. Concrete agent architectures, logical agents, introduction to Bayesian reasoning **[7]**
- Introduction to real-time systems, real-time environment, distributed solutions, notion of time, fault-tolerance, modeling real-time systems **[4]**

Fractal II: Design of Autonomous vehicles [14 Lectures]

- Architecture of autonomous vehicles, processing pipeline, Vehicle Networks **[1]**
- Sensors for Navigation, GPS, IMU, Lidar, Camera, Visual Odometry, Place Recognition, Extraction based on Range Data **[5]**
- Localization, Noise and Aliasing, Belief Representation, Map Representations, Probabilistic Map based localization, Autonomous Map Building, SLAM **[4]**
- Planning and Reacting, Path Planning and Obstacle Avoidance, Navigation Architectures, Data Fusion **[4]**

Fractal III: Multi-agent systems [14 Lectures]

- Multi-agent systems and organizations, Hierarchical systems & peer-to-peer organization, Openness & scalability, Emergent organization **[2]**
- Communication and coordination in distributed systems, Distributed Planning and Execution, Task sharing, Coordination without communication [3]
- Game Theory, 2-player games, Nash Equilibrium, Strategy, Adversarial search, Iterative game and learning, Pareto-efficiency, Negotiation, Social choice, Mechanism design, Auction **[7]**
- Autonomous systems and human society, ethics and trust in multi-agent systems, explainability **[2]**

Text Books

1. Gerhard Weiss ed. (2013), Multiagent System, Second Edition, MIT Press.
2. Roland Siegwart, Illah Reza Nourbakhsh, Davide Scaramuzza (2018), Introduction to autonomous mobile robots, MIT press.

Reference Books

1. Michael Wooldridge (2009), An introduction to multi-agent systems (2nd ed). Wiley
2. Stuart J. Russell and Peter Norvig (2019), Artificial Intelligence: A Modern Approach, 4th edition,Pearson Press
3. Hermann Kopetz. (2011). Real-time Systems (2nd ed). Kluwer Academic Publishers
4. Choset, H. M., Hutchinson, S., Lynch, K. M., Kantor, G., Burgard, W., Kavraki, L. E., &Thrun, S. (2005). Principles of robot motion: theory, algorithms, and implementation. MIT press.

Other study materials

Research papers: (to be shared in class)

Online Course Material:

https://www.edx.org/course/autonomous-mobile-robots

Title	**Bio-image Computing**	Number	CSL7382
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech, M.Tech., PhD	Type	Elective
Prerequisite	Computer Programming, Linear algebra, Probability and Statistics	Antirequisite	None

Objectives
The Instructor will:
1. Provide details of bio-signal and medical image acquisition process
2. Explain information extraction and image analysis techniques using machine learning with emphasis on the field of healthcare, agriculture and environment

Learning Outcomes
The students are expected to have the ability to:
1. Understand different imaging modalities and acquisition process
2. Apply machine learning techniques for bio-signal interpretation, image representation and analysis

Contents
CSL7XX1 Bio-signal Acquisition and Representation 1-0-0 [1]
Introduction: Overview of biological signals and biomedical imaging modalities, ECG, NMR spectroscopy, electron microscopy, magnetic resonance imaging, X-ray, computed tomography, positron emission tomography, ultrasound, elastography, optical imaging and others, Noise and error propagation in biomedical signals and image data (10 lectures)
Visualization: Sectioning, multimodal images, overlays, rendering surfaces and volumes (4 lectures)

CSL7XX1 Machine Learning for Bio-signal analysis 1-0-0 [1]
Reconstruction: Mathematical models of image regularity, random fields, practical data sampling and acquisition schemes (4 lectures)
Restoration: Deconvolution, degradation models for corrupted and missing data, Bayesian graphical modeling and inference, regression methods for filtering of CT, MRI ultrasound and other images (4 lectures)
Image segmentation, object delineation, classification: Clustering, graph partitioning, classification, mixture models, expectation maximization, variational methods using geometric and statistical modeling, computer aided diagnosis (4 lectures)
Registration: Deformation models, optimization algorithms, 2D-3D registration, multi-modal registration (2 lectures)

CSL7XX2 Deep Learning for Bio-imaging 1-0-0[1]
Enhancement, Segmentation of anatomical structures, subcellular objects, cells, learning with little or no training data, spatial transformer network for registration, image-based phenotyping, analysis of radio-genomic data (10 lectures)
Analysis of motion: Tracking of cells, tissues, organisms, and particles (2 lectures)
Interactive image analysis: Human in loop, image interpretation (2 lectures)

Laboratory Experiments
Ultrasound image enhancement, tumor segmentation in BraTS dataset, registration of MRI images etc.

Textbook
1. Wu, G., (2016), *Machine Learning and Medical Imaging*, Elsevier.
2. Epstein, C.L., (2003), *Mathematics of Medical Imaging*, Prentice Hall.
3. Bankman, I., (2009), *Handbook of Medical Image Processing and Analysis*, 2nd Edition, Academic Press.

Preparatory Course Material
1. Bishop, C., (2006), *Pattern Recognition and Machine Learning,* Springer.

Title	**Blockchain**	Course No.	CSL 7XX
Department	Computer Science and Engineering	Structure (L-T-P [C])	3–0–0 [3]
Offered for	B.Tech, MTech, PhD.	Type	Elective
Prerequisite	Cryptography	Antirequisite	

Objectives
The Instructor will:
1. Explain how blockchain technology works
2. Integrate blockchain technology into the current business processes to make them secure

Learning Outcomes

1. Understand what and why of Blockchain
2. Explore major components of Blockchain and Identify a use case for a Blockchain application Create their own Blockchain network application

Contents
Introduction to Blockchain: Digital Trust, Asset, Transactions, Distributed Ledger Technology, Types of network, Components of blockchain, mining, double spending attack, 51% attack. (5 Lectures)
PKI and Cryptography: Private keys cryptography, Public keys cryptography, Hashing, Digital Signature. (6 Lectures)
Consensus: Byzantine Fault, Proof of Work, Proof of Stake. (5 Lectures)
Cryptocurrency: Bitcoin creation and economy, Limited Supply and Deflation, Hacks, Ethereum concept and Ethereum classic, Monero, Zcash, Zero knowledge proofs, etc. (9 Lectures)
Permissioned blockchains: Permissionless versus permissioned Blockchains (2 Lectures)
Hyperledger Fabric: Hyperledger Architecture, Membership, Blockchain, Transaction, Chaincode, Hyperledger Fabric, Features of Hyperledger, Fabric Demo. (8 Lectures)
Blockchain Applications: Building on the Blockchain, Ethereum Interaction - Smart Contract and Token (Fungible, non-fungible), Languages,Payment escrow, Micro payments, Decentralized lotteries Blockchain-as-a-service. (6 Lectures)

Textbook
A. BAHGA, V. MADISETTI (2017), Blockchain Applications: A Hands-On Approach, VPT.

Self Learning Material
1. M. SWAN (2015), Blockchain: Blueprint for a New Economy, O'Reilly Media.
2. R. WATTENHOFER (2016), The Science of the Blockchain, CreateSpace Independent Publishing Platform.
3. I. BASHIR (2017), Mastering blockchain, Packt Publishing Ltd.
4. K.E. LEVY, Book-smart, Not Street-smart: Blockchain-based Smart Contracts and the Social Workings of Law, Engaging Science, Technology, and Society, vol. 3, pp. 1-15, 2017.

Preparatory Course Material
MIT Online Blockchain Course, Learn Blockchain Technology: https://getsmarter.mit.edu/

Course Title	**Compiler Design**	Course No.	**CSL7xx0**
Department	Computer Science and Engineering	Structure (L-T-P- D[C])	3-0-0-0 [3]
Offered for	B.Tech., M.Tech. Ph.D.	Type	Elective
Prerequisite	Maths for Computing	Antirequisite	

Objectives
1. To introduce key challenges for modern compilers and runtime systems
2. To explain various optimization techniques of a typical compilation workflow.

Learning Outcomes
1. To learn about compilation techniques for obtaining high performance on modern computer architectures.
2. To analyze and optimize various components of compilation stages

Contents
Representing Programs: Abstract Syntax Tree, Control Flow Graph, Dataflow Graph, Static Single Assignment, Control Dependence Graph, Program Dependence Graph, Call Graph. (11 Lectures)
Analysis/Transformation Algorithms: Dataflow Analysis, Interprocedural analysis, Pointer analysis, Rule-based analyses and transformations. Constraint-based analysis. (12 Lectures)
Applications: Scalar optimizations, Loop optimizations, Object-oriented optimizations, Register allocation, Program verification, Bug finding. (8 Lectures)
Advanced Topics: Just-in-time compilation, Memory Management, EDGE architecture compilation, Power- aware compilation, Machine specific optimizations (11 lectures).

Text Books
K.D. COOPER, L.TORCZON (2011), Engineering a Compiler, Morgan Kaufmann, 2nd Edition.
S. MUCHNICK (1998), Advanced Compiler Design Implementation, Morgan Kaufmann.

Reference Books
1. Y.N. SRIKANT, P.SHANKAR (Eds.) (2007), The Compiler Design Handbook: Optimizations and Machine Code Generation, CRC Press, 2nd Edition.
2. A.V. AHO, M.S. LAM, R. SETHI, J.D. ULLMAN (2007), Compilers: Principles, Techniques and Tools, Addison-Wesley, 2nd Edition.
3. R. ALLEN, K. KENNEDY (2001), Optimizing Compilers for Modern Architectures: A Dependence Based Approach, Morgan Kaufmann, 1st Edition.
4. M. WOLFE (1995), High-Performance Compilers for Parallel Computing, Addison-Wesley.
5. IEEE and ACM Digital Library.

Self Learning Material
NPTEL Lecture: https://nptel.ac.in/courses/106/108/106108052/

Course Title	**Complexity Theory**	Course No.	**CSL7140**
Department	Computer Science and Engineering	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B.Tech CSE, M.Tech CSE, Ph.D.	Type	Elective
Prerequisite	Maths for Computing, Data Structures and Algorithms	Antirequisite	

Objectives
The course provides a challenging introduction to some of the central ideas of theoretical computer science. It attempts to present a vision of computer science beyond computers.

Learning Outcomes
Students are expected to understand:
1. The theory of classical complexity classes and its challenges.
2. The state-of-the-art in the field of complexity theory.

Contents
Basic Complexity Classes: The computational model, NP and NP completeness, Diagonalization, Space complexity, The Polynomial hierarchy and alternations, Circuits, Randomized computation, Interactive proofs, Complexity of counting, Complexity issues in cryptography. (16 Lectures)
Lower Bounds for Concrete Computational Models: Decision trees, Communication complexity, Circuit lower bounds, Algebraic computation models. (10 Lectures)
Advanced topics: Average case complexity: Levin's theory, Derandomization, Expanders and extractors, Hardness amplification and Error correcting codes, PCP and hardness of approximation, Logic in complexity theory, Quantum computation. (16 Lectures)

Text Books
S. ARORA, B. BARAK (2009), Computational Complexity: A Modern Approach, Cambridge University Press, 1st Edition.

Self Learning
MIT Lecture Notes, Automata, Computability, and Complexity, Electrical Engineering and Computer Science:
https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-045j-automata-computability-and-complexity-spring-2011/lecture-notes/

Title	**Computational Microeconomics**	Number	CS L7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech, M.Tech	Type	Elective
Prerequisite	Design and Analysis of Algorithms	Antirequisite	None

Objectives
The Instructor will:
1. Familiarize with the market design
2. Focus on algorithms to clear the market
3. Familiarize with the algorithmic game theory

Learning Outcomes
The students will have the ability to:
1. Design market
2. Use computer science knowledge to solve economic problems

Contents
Introduction to Market Design: introduction to various types of auctions, bidding languages, matching market, voting. [6 lectures]
Auction: Myerson’s Lemma, Knapsack auction, revenue-maximising auction, exchanges. [8 lectures]
Matching Market: Kidney exchange, stable matching, various notions of optimal matching. [8 lectures]
Noncooperative Game Theory: self-interested agents, price of anarchy, non-atomic and atomic models of selfish routing. [5 lectures]
Games in Normal Form: pareto optimality, nash equilibria, various solution concepts and their computation. [10 lectures]
Games in Extensive Form: backwards induction, subgame perfect equilibrium, imperfect information, equilibrium refinements, computing equilibria. [5 lectures]

Text Book
Shoham, Y. & Brown, K.L., (2008), Multiagent systems: Algorithmic, game-theoretic, and logical foundations, Cambridge University Press.

Reference Books
Roughgarden, T. (2016), Twenty lectures on algorithmic game theory, Cambridge University Press.

Self Learning Material
https://www.youtube.com/watch?v=TM_QFmQU_VA

Title	**Computational Optimization**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech, M.Tech., Ph.D.	Type	Elective
Prerequisite	Data Structures and Algorithms	Antirequisite	None

Objectives
The Instructor will:
1. Introduce various terminologies, concepts and algorithms related to classical, heuristic and nature inspired optimization algorithms
2. Discuss their applications in real-world scenarios

Learning Outcomes
The students are expected to have the ability to:
1. Utilize state of the art heuristic optimization algorithms in their research activities
2. Design and propose new and hybrid optimization algorithms
3. Customize heuristic optimization algorithms for special applications

Contents
Introduction, Definitions and Concepts: Optimization, Operational Research (OR), Engineering Optimization, Definition of an Optimization Problem, Feasibility Problem, Classification of Optimization Problems, Classification of Optimization Techniques, Heuristic Algorithms vs. Metaheuristics, Swarm Intelligence, Population-Based Optimization, Multi-objective Optimization, Parallelization, Evaluation of the Optimization Algorithms (6 lectures)
Overview of Classical Optimization Techniques: Linear programming, Nonlinear Programming (3 lectures)
Overview of Heuristic Optimization Algorithms: Neighborhood Search, Hill Climbing Methods, Greedy Algorithms, Simulated Annealing (3 lectures)
Overview of Nature Inspired Optimization Algorithms: Evolutionary Algorithms, Tabu Search, Ant Colony Optimization, Particle Swarm Optimization (2 lectures)
Simulated Annealing: Real Annealing and Simulated Annealing, Metropolis Algorithm, Simulated Annealing Algorithm, Continuous Simulated Annealing, One-loop Simulated Annealing, Temperature Scheduling, Convergence of Simulated Annealing, Applications, Normalization of the Parameters, Tuning the Parameters of an algorithm (9 lectures)
Evolutionary Algorithms: Methods of encoding, Operators of Evolution, Models of Evolution, Genetic Algorithms, Steady State Genetic Algorithms, Genetic Programming, Memetic Algorithms, Differential Evolution (7 lectures)
Tabu Search: Basic Tabu Search, Short-term Memory, Long-term Memory, Diversification and Intensification, Continuous Tabu Search (4 lectures)
Ant Colony Optimization: Collective Behavior of Social Insects, Basic ACO Algorithms, Ant Algorithms for TSP, Adaptation to Continuous Problems, Applications (5 lectures)
Particle Swarm Optimization: Canonical PSO Algorithm, Important Parameters, Neighborhood Topologies (3 lectures)

Textbook
1.Michalewicz,Z. and Fogel,D.B., (2004), *How to Solve it: Modern Heuristics*, 2nd Edition, Springer
2.Simon, D., (2013), *Evolutionary Optimization Algorithms*, Wiley
3.Yang,X.S., (2014), *Nature-inspired Metaheuristic Algorithms*, Luniver Press

Reference Books
1. Rao, S.S., (2013), *Engineering Optimization: Theory and Practice*, 3rd Edition, New Age International Publishers

Self Learning Material
1. NPTEL: Traditional and Non-traditional Optimization Tools
https://nptel.ac.in/courses/112105235/1

Title	**Computational Social Choice Theory**	Number	CS L7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech, M. Tech, Ph.D.	Type	Elective
Prerequisite	Design and Analysis of Algorithms	Antirequisite	None

Objectives

The Instructor will:
1. Familiarize with mechanisms for collective decision making
2. Focus on interface of social choice theory with theoretical computer science
3. Enable students to conduct research in this fast-growing field

Learning Outcomes

Students will have the ability to:
1. Understand the problems and research directions in computational social choice.
2. Mathematically formulate the problems that arise in society, and design efficient rules for these problems.
3. Analyse computational complexity of problems in social choice.

Contents

Introduction to Voting Theory: social choice functions, voting rules and their axiomatic study. *[5 Lectures]*
***Computational hardness and Algorithms for voting rules*:** Condorcet, minimax, Dodgson's rule, and Kemeny. *[5 lectures]*
Manipulation in Voting: various means of manipulation: strategic voting, bribery, control, computational hardness and algorithms. *[7 lectures]*
Multiwinner Election: voting rules and their computational complexity. *[4 lectures]*
Matching under preferences: introduction to stable matching, Gale-Shapley algorithm, hospital/resident problem, stable roommate problem. *[8 lectures]*
Manipulation in Matching: means of manipulations, computational hardness and algorithms. *[3 lectures]*
Introduction to Fair Allocation: basic terminologies, fairness and efficiency criteria. *[4 lectures]*
Fair Allocation of goods: fair allocation of indivisible and divisible goods. *[6 lectures]*

Text Book

1. Brandt, F., Conitzer, V., Endriss, U., Lang, J., & Procaccia, A.D. (2016). Handbook of Computational Social Choice. Cambridge University Press.
2. Gusfield, D., & Irving, R.W. (1989). The Stable Marriage Problem: Structure and Algorithms. MIT press
3. Manlove, D. (2013). Algorithmics Of Matching Under Preferences. World Scientific

Reference Books

1. Endriss, U. (2017). Trends in Computational Social Choice. Lulu. com
2. Lecture Notes on Fair Division by U. Endriss

Title	**Computer Graphics (700)**	Number	CSL7450
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech, M.Tech., Ph.D.	Type	Elective
Prerequisite	None	Antirequisite	Computer Graphics (400) - CSL4xx

Objectives

The Instructor will:

1. Provide a thorough introduction to computer graphics techniques, focusing on 2D and 3D modelling, image synthesis and rendering

Learning Outcomes

The students are expected to have the ability to:

1. Explain and create interactive graphics application
2. Implement graphics primitives
3. Synthesize and render images for animation and visualization

Contents

CSL7xx1 Introduction to Graphical Primitives 1-0-0 [1]

Introduction: Overview of computer graphics, representing pictures, preparing, presenting & interacting with pictures for presentations; Scan conversion: 2D Geometric Primitives; Area Filling algorithms. Clipping algorithms, Anti Aliasing

Transformations and viewing: 2D and 3D transformations, Matrix representations & homogeneous coordinates, Viewing pipeline, Window to viewport coordinate transformation, clipping operations, viewport clipping, 3D viewing.

CSL7xx2 Graphical Object Representation 1-0-0 [1]

Curves and Surfaces: Conics, parametric and non-parametric forms; Bezier (Bernstein Polynomials) Curves, Cubic-Splines, B-Splines; Quadratic surfaces, Bezier surfaces and NURBS, 3-D modelling.

CSL7xx3 Graphics Rendering 1-0-0 [1]

Hidden surfaces: Depth comparison, Z-buffer algorithm, Back face detection, BSP tree method, the Printer's algorithm, scan-line algorithm; Hidden line elimination, wire frame methods, fractal - geometry.

Color & shading models: Phong's shading model, Gouraud shading, Shadows and background, Color models, Photo-realistic rendering, Animation and OpenGL primitives: Functions, pipeline, sample programs for drawing 2-D, 3-D objects; event handling and view manipulation, Introduction to GPU and animation

Textbook

1. J. D. Foley, A. Van Dam, S. K. Feiner and J. F. Hughes, Computer Graphics; Principles and practice, Addison Wesley, 2nd Edition in C, 1997.
2. D. F. Rogers and J. A. Adams, Mathematical elements for Computer Graphics, McGraw-Hill, 2nd Edition, 1990.

Self-Learning Material

1. Blender: *https://www.blender.org/download/*
2. OpenGL: http://www.opengl-tutorial.org/

Preparatory Course Material

1.Department of Computer Science and Engineering, Indian Institute of Technology Madras, https://nptel.ac.in/courses/106106090/

Title	**Computer Network Protocols and Applications**	Number	**CS7xx**
Department	Computer Science	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech, M.Tech., Ph.D.	Type	Elective
Prerequisite	Data Structures and Algorithms	Antirequisite	Computer Networks

Objectives

1. To understand the organization of computer networks, factors influencing the performance of computer networks, and the reasons for having a variety of different types of networks.
2. To understand the Internet structure, various protocols of the Internet and how these protocols address the standard problems of networking and the Internet.
3. To introduce some advanced concepts both in the wired as well as wireless networks.

Learning Outcomes

1. Familiarity with the essential protocols of computer networks in terms of design, implementation and operations.
2. Students would be able to appreciate different types of wired and wireless networks.

Contents:

Introduction: Review of the Internet architecture and layering (2 Lectures)

Application Layer: Application layer protocols, End-to-End data, multimedia networking applications (image, audio, video compression) (6 Lectures)

Transport Layer: TCP congestion and flow control, Flow Rate Fairness Algorithms, Fair Queueing Algorithms Congestion Avoidance and Control, Algorithms for Congestion Avoidance in Computer Networks , Congestion Control for High Bandwidth-Delay Product Networks (8 lectures)

Network Layer: End-to-End Routing Algorithm, intra- and inter-domain routing Algorithms, e.g., BGP, MPLS, Stable Paths Problem and Selfish Routing Algorithms, Mobility aspects and MobileIP, Routing in Ad hoc and wireless networks (8 Lectures)

Software defined networks and Network Virtualization: OpenFlow/SDN control, Mininet, Network **Function Virtualization, Networking in dockers**. (6 Lectures)

Advanced topics: VPN, Data centre Networking, Content Delivery Networks, Overlay networks, low power wide range communication, TV Whitespaces (12 lectures)

Text Book

1. K.W. ROSS, J.F. KUROSE (2016), Computer Networks: A Top Down Approach, Pearson, 7th Edition.

Reference Books

1. W. STALLINGS (2013), Data and Computer Communications, Pearson, 10th Edition.
2. L.L. PETERSON, B.S. DAVIE (2020), Computer Networks: A Systems Approach, Morgan Kaufmann, 6th Edition.

Supplementary Resources:

1. TCP/IP Tutorial and Technical Overview, (IBM Redbook) - Download From http://www.redbooks.ibm.com/abstracts/gg243376.html
2. TCP/IP Guide, Charles M. Kozierok, Available Online - http://www.tcpipguide.com/
3. NPTEL Online Lectures: https://onlinecourses.nptel.ac.in/noc21_cs18/preview
4. Selected research papers

Title	**Computer Vision**	Course No.	CSL7xx
Department	Computer Science and Engineering, AI&DS	Structure (L-T-P [C])	3–0–0 [3]
Offered for	B.Tech., M.Tech., Ph.D.	Type	Elective
Prerequisite	Linear Algebra		

Objectives

The Instructor will:

1. Provide insights into fundamental concepts and algorithms behind some of the remarkable success of Computer Vision
2. Impart working expertise by means of programming assignments and a project

Learning Outcomes

The students are expected to have the ability to:

1. Learn and appreciate the usage and implications of various Computer Vision techniques in real-world scenarios
2. Design and implement basic applications of Computer Vision

Contents

Introduction*:* The Three R's - Recognition, Reconstruction, Reorganization (1 Lecture)
Fundamentals: Formation, Filtering, Transformation, Alignment, Color (5 Lectures)
Image Restoration*:* Spatial Processing and Wavelet-based Processing (5 Lectures)
Geometry: Homography, Warping, Epipolar Geometry, Stereo, Structure from Motion, Optical flow (9 Lectures)
Segmentation: Key point Extraction, Region Segmentation (e.g., boosting, graph-cut and level-set), RANSAC (6 Lectures)
Feature Description and Matching: Key-point Description, handcrafted feature extraction (SIFT, LBP) (3 Lectures)
Deep Learning based Segmentation and Recognition: DL-based Object detection (e.g. Mask-RCNN, YOLO), Semantic Segmentation, Convolutional Neural Network (CNN) based approaches to visual recognition (9 Lectures)
Applications: Multimodal and Multitask Applications (4 Lectures)

Textbooks

1. R. HARTLEY, A. ZISSERMAN (2004), Multiple View Geometry in Computer Vision, Cambridge University Press, 2nd Edition.
2. R. SZELISKI, (2010), Computer Vision: Algorithms and Applications, Springer-Verlag London.

Reference Books

1. Research literature

Title	**Crowd-sourcing and human-computing**	Number	CS L7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech (All), M.Tech	Type	Elective
Prerequisite	Data Structures & Algorithms	Antirequisite	None

Objectives
The Instructor will:
1. familiarize with the techniques to manage the crowd
2. focus on building intelligent systems that involve a combination of computers and humans collaborating seamlessly over the Internet

Learning Outcomes
The students will have the ability to:
1. Program the crowd
2. Apply usability principles for designing crowd tasks that elicit high-quality responses
3. Use statistical methods to improve the quality of the work received
4. Build systems that interface with crowd labor in real time

Contents
Introduction to Crowdsourcing and Human Computation: introduction, history, connections to social computing, collective intelligence, and management science [4 lectures]
Human Computation Algorithms: design patterns, case study, discussion [4 lectures]
Usability Engineering for CrowdSourcing: input-agreement mechanisms, output-agreement mechanisms [4 lectures]
Gold Standards: consensus algorithms, incentive mechanisms, gamification [6 lectures]
Real-time crowdsourcing: queuing theory for predicting worker availability, ensemble learning, crowd agents [6 lectures]
Crowdsourcing Subjective Tasks: social Q&A, expertise in crowdsourcing, local crowdsourcing / community sourcing, crowd funding, citizen science [6 lectures]
Task Routing: push and pull approaches [4 lectures]
Crowds & Applications: Crowds: mechanical turk, social media; Applications: computer vision, NLP [8 lectures]

Text Book
Law, E., & Ahn, Ahn, L. (2012). Human Computation: An Integrated Approach to Learning from the Crowd, Morgan & Claypool Publishers.

Reference Books
1. Ghosh, A., Lease, M., (2016). Human Computation and Crowdsourcing. Proceedings of the Fourth AAAI Conference
2. Lease M., Alonso O. (2018) Crowdsourcing and Human Computation: Introduction. In: Alhajj R., Rokne J. (eds) Encyclopedia of Social Network Analysis and Mining. Springer, New York.

Self-Learning Material

Title	**Cryptography**	Number	**CSL 7480**
Department	Computer Science	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech., M.Tech., Ph.D.	Type	Elective
Prerequisite	None	Antirequisite	

Objectives
Introduce students to the domain of Symmetric and Asymmetric Cryptography, and develop an appreciation for some applications in which Cryptography is used in real-life.

Learning Outcomes
1. Learn to model security of various cryptographic primitives as an adversarial game.
2. Learn the design principles of Block ciphers, stream ciphers, hash functions, MAC's, public key encryption, and digital signatures.
3. Identify novel and significant open research questions in the field.
4. Using cryptographic primitives to design some real life protocols, such as ZKP, coin tossing over insecure-channel, and MPC.

Contents:
Foundations [9 lectures]: Attacking Toy ciphers, Kerckhoff's law, Perfect security, OWF, PRG, PRF, PRP
Symmetric key cryptography [12 lectures]: Block ciphers, stream ciphers, hash functions, MAC, authenticated encryption.
Asymmetric Key cryptography [12 lectures]: DKHE, RSA, El Gamal cryptosystems (with required mathematical foundations), Elliptic Curve Cryptography, Digital signatures, Public Key Infrastructure and Digital certificates.
Cryptographic protocols [9 lectures]: Secure multiparty computation, Zero knowledge proofs, Oblivious transfer etc.

Textbooks
1. Katz, Lindell: Theory of modern cryptography, CRC Press, 3rd Edition.
2. Hoffstein, Pfipher, Silverman: Introduction to mathematical cryptography

Self Learning Material
1. https://nptel.ac.in/courses/106106221

Course Title	**Cybersecurity**	Course No.	**CS 7XXX**
Department	Computer Science and Engineering	Structure (L-T-P [C])	3-0-2 [4]
Offered for	B. Tech, MTech, Ph. D.	Type	Elective
Prerequisite	Computer Networks	Anti-req	Security and its applications

Objectives

1. The Instructor will provide the skills needed to protect networks, secure electronic assets, prevent attacks, ensure the privacy of your customers, and build secure infrastructure

Learning Outcomes

1. To protect data and respond to threats that occur over the Internet
2. Design and implement risk analysis, security policies, and damage assessment
3. An ability to apply security principles and practices to the environment, hardware, software, and human aspects of a system.

Contents:

Introduction to Cyber Security: *Internet governance – Challenges and constraints, Cyber threats. (2 Lectures)*

Cyber Security Vulnerabilities and Cyber Security Safeguards: *Cyber security, Vulnerabilities, safeguards, Access control, Authentication, Biometrics, Deception, Denial of Service Filters, Ethical hacking, Firewalls, Response, Scanning, Security policy, Threat management. (8 Lectures)*

Cryptography: *Shannon's Approach to Cryptography: Measures of security, Perfect secrecy, Definition of entropy, One-time pad , Symmetric Key Cryptography, Cryptographic Hash Functions, Authentication, Public Key Cryptosystems, Key Distribution and Key Agreement Protocols(4 Lectures)*

Securing Web Application, Services and Servers: *Basic security for HTTP applications and services, Basic security for SOAP services, Identity management and Web services, Authorization patterns, Security considerations, challenges. (4 Lectures)*

Network Security: *TCP/IP threats, The IPSEC protocol, The SSL and TLS protocols, Firewalls and Virtual Private Networks (VPNs), Electronic mail security, Worms, DDoS attacks, BGB and security considerations (12 lectures)*

Intrusion Detection and Prevention: *Intrusion detection and Prevention techniques, Anti-malware software, Security information management, Network session analysis, System integrity validation. (4 Lectures)*

Cyberspace and the Law: *Cyber security regulations, Roles of international law, State and Private Sector in cyberspace, Cyber security standards. The INDIAN cyberspace, National cyber security policy. (5 Lectures)*

Cyber Forensics: *Handling preliminary investigations, Controlling an investigation, Conducting disk-based analysis, Investigating information-hiding, Tracing internet access, Tracing memory in real-time. (4 Lectures)*

Laboratory

1. Design and implementation of a simple client/server model and running application using sockets and TCP/IP.
2. To make students aware of the insecurity of default passwords, printed passwords and passwords transmitted in plain text.
3. To teach students how to use SSH for secure file transfer or for accessing local computers using port forwarding technique.
4. Comparison between Telnet and SSH for Secure Connection
5. AVISPA Tool for the Automated Validation of Internet Security Protocols and Applications

Text Book

1. C.J. HOOFNAGLE (2016), Federal Trade Commission Privacy Law and Policy, Cambridge University Press, 2016.
2. Stallings, W. (2017). Cryptography and network security, 7/E. Pearson Education India

Reference Book

1. Douglas R. Stinson, Maura B. Paterson (2018). Cryptography: theory and practice. 4/E Chapman and Hall/CRC
2. P.W. SINGER, A. FRIEDMAN (2014), Cybersecurity: What Everyone Needs to Know, OUP, 1st Edition.

Supplementary Resource

Title	**Data Visualization (700)**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech, M.Tech., PhD	Type	Elective
Prerequisite	Data Engineering / DBMS	Antirequisite	Data Visualization (400) - CSL4xx

Objectives
The Instructor will:
1. Provide background to understand various aspects of Data Visualization
2. Discuss various principles of visualizing heterogeneous types of data

Learning Outcomes
The students are expected to have the ability to:
1. Present data with visual representations for the target audience, task, and data
2. Analyze, critique, and revise data visualizations
3. Apply appropriate design principles in the creation of presentations and visualizations

Contents
Visual Queries: Process of Seeing, The Act of Perception, Design Implications, Distributed Cognition, Visual Search Strategies (3 lectures)
Data and Visualization: Data Type, Coordinate Systems, Scale (2 lectures)
Visualization Design: Amount, Distribution, Proportion, Trends, Time Series, Geospatial (10 lectures)
Narratives: Telling Stories with Data, Sequencing, Visualization Rhetoric, text visualization (4 lectures)
Mapping and Cartography: The Cartogram, Value-by-Area Mapping (4 lectures)
Optimal Space Usage: Aspect Ratio Selection, Geometry & Aesthetics, Wilkinson's Algorithm and its extension (6 lectures)
Networks: Scalable, Versatile and Simple Constrained Graph Layout, Visualization of Adjacency, Multiple Network Analysis and Visualization, Visualizing Online Social Networks (7 lectures)
Animation and Color: Trend Visualization, Transitions in Statistical Data Graphics, Graphs with Radial Layout, Cartoons, Color and Information, Infographics (7 lectures)

Textbook
1. Tufte,E., (2001), *The Visual Display of Quantitative Information*, 2nd Edition, Graphics Press
2. Tufte,E., (1990), *Envisioning Information*, Graphics Press

Self-Learning Material
1. Wilke,C.O., (2019), *Fundamentals of Data Visualization: A Primer on Making Informative and Compelling Figures*, O'Reilly Media
2. Ware,C. and Kaufman,M., (2008), *Visual thinking for design*. Burlington: Morgan Kaufmann Publishers
3. Wong,D., (2011), *The Wall Street Journal guide to information graphics: The dos and don'ts of presenting data, facts and figures*, New York: W.W. Norton & Company

Preparatory Course Material
1. Data Visualization Course, https://curran.github.io/dataviz-course-2018/

Title	**Digital Image Analysis**	Number	CSLXXX
Department	Computer Science and Engineering, EE	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech, M.Tech., PhD	Type	Elective
Prerequisite	Linear Algebra	Antirequisite	None

Objectives

The Instructor will:

1. To introduce the origin and formation of digital imaging.
2. To develop the understanding of different types of imaging techniques for different purposes.
3. To equip the students with various possible applications of image analysis.

Learning Outcomes

Students are expected to have the ability to:

1. Enhance image in spatial and frequency domain.
2. Implement various aspects of image segmentation and compression.

Contents

Digital Image Fundamentals: Image modeling, Sampling and Quantization, Imaging Geometry, Digital Geometry, Image Acquisition Systems, Different types of digital images (3 Lectures)
Image Transforms: Basic transforms: Spatial and Frequency Domain Transforms (8 Lectures)
Image Enhancement: Point processing, interpolation, enhancement in spatial domain, enhancement in frequency domain (7 Lectures)
Color Image Processings:Color Representation, Laws of color matching, chromaticity diagram, color enhancement, color image segmentation, color edge detection (3 Lectures)
Image compression: Lossy and lossless compression schemes, prediction based compression schemes,
vector quantization, sub-band encoding schemes, JPEG compression standard (4 Lectures) Morphology:Dilation, erosion, opening, closing, hit and miss transform, thinning, extension to grayscale morphology, Euler technique (5 Lectures)
Segmentation: Segmentation of grey level images, Watershed algorithm for segmenting grey level image (6 Lectures)
Feature Detection:Fourier descriptors, shape features, object matching/features (6 Lectures)

Textbook

1. C. GONZALEZ, R.E. WOODS (2018), Digital Image Processing, Prentice Hall, 4th Edition.
2. A.K. JAIN (1989), Fundamentals of Digital Image Processing, Prentice Hall.

Self-Learning Material

https://nptel.ac.in/courses/117104020/

Course Title	**Distributed Algorithms**	Course No.	**CS7xxx**
Department	Computer Science and Engineering	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B.Tech, M. Tech, Ph.D.	Type	Bouquet/Elective
Prerequisite	Design and Analysis of Algorithms	Antirequisite	None

Objectives
The objective of the course is to introduce the distributed algorithms in real-world systems

Learning Outcomes
Students will be able to:
1. Introduce the most important basic results in the area of distributed algorithms, and prepare interested students to begin independent research or take a more advanced course in distributed algorithms.
2. Implement distributed algorithms in real-world distributed computing platforms
3. Understand and identify applications of distributed algorithms in real-world systems

Contents
Introduction to Distributed Computing Models : Model of a distributed system: asynchronous message-passing model, Time and message complexity, Safety and liveness properties, Clock Synchronization [5 Lectures]
Leader Election Logical Clocks and Causality: *Leader election in rings, Asynchronous leader election with identities,*
Synchronous leader election by abusing the synchronous model, Logical clocks and vector clocks , Snapshots and synchronization [7 Lectures]
Message Ordering and Group Communication: Termination Detection Algorithms and Reasoning with Knowledge [4 Lectures]
Distributed Mutual Exclusion Algorithms: *Lamport's algorithm, Ricart–Agrawala algorithm, and Deadlock Detection Algorithms* [6 Lectures]
Global Predicate Detection: Modalities on predicates, Centralized algorithm for relational predicates,
Conjunctive predicates, Distributed algorithms for conjunctive predicates [3 Lectures]
Distributed Shared Memory: Memory consistency models, Shared memory mutual exclusion and Wait- freedom [6 Lectures]
Checkpointing and Rollback Recovery: Checkpoint-based recovery and Log-based rollback recovery [3 Lectures]
Consensus and Agreement : Agreement in a failure-free system (synchronous or asynchronous) , Agreement in (message-passing) synchronous systems with Failures, Agreement in asynchronous message-passing systems with failures , Wait-free shared memory consensus in asynchronous systems [6 Lectures]
Distributed Programming Environments: Communication primitives, selected case studies [2 Lectures]

Text book
A.D. KSHEMKALYANI, M. SINGHAL (2011), Distributed Computing: Principles, Algorithms, and Systems, Cambridge University Press.

Reference Books
1. N. LYNCH, (1996), Distributed Algorithms, Morgan Kaufmann, 1st Edition.
2. G. TEL, (2000), Introduction to Distributed Algorithms, Cambridge University Press, 2nd Edition.

Course Title	**Distributed Database Systems**	Course No.	**CS7XX**
Department	Computer Science & Engineering	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B.Tech, MTech, PhD	Type	Elective
Prerequisites	Operating Systems, Database Systems, Computer Networks	Antirequisite	None

Objective
To understand and appreciate concepts of distributed database design, and its associated issues of consistency, concurrency, optimization, integrity, reliability, privacy, and security.

Learning Outcome
Ability to understand the need for distributed database systems and its related complexities pertaining to fragmentation, replication, availability, concurrency, consistency and recovery.

Contents
Introduction: Distributed data processing concepts, What is a DDBS - advantages, disadvantages and problem areas. (2 Lectures)
Distributed Database Management System Architectures: Transparencies, Architecture, Global directory concepts and issues. (3 Lectures)
Distributed Database Design: Design strategies, Design issues, Fragmentation, Data allocation. (4 Lectures)
Semantics Data Control: View management, Data security, Semantic integrity control. (5 Lectures)
Query Processing: Objectives, Characterization of processors, Layers of processing, Query decomposition, Data localization. (5 Lectures)
Query Optimization: Factors, Centralized query optimization, Fragmented query ordering, Query optimization algorithms. (5 Lectures)
Transaction Management: Goals, Properties, Models. (4 Lectures)
Concurrency Control: Concurrency control in centralized systems, Concurrency control in DDBSs - algorithms, Deadlock management. (5 Lectures)
Reliability: Issues and types of failures, Reliability techniques, Commit protocols, Recovery protocols. (5 Lectures)
Other Avenues: Parallel database Systems, Multi-databases. (4 Lectures)

Reference Books
S. CERI, G. PELAGATTI (2008), Distributed Databases: Principles and Systems, McGraw-Hill, 1st Edition (2017 Reprint).
M.T. ÖZSU, P. VALDURIEZ (2011), Principles of Distributed Database Systems, Springer, 3rd Edition.

Course Title	**Edge and Fog Computing**	Course No.	**CSL7xx**
Department	Computer Science & Engineering	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B.Tech	Type	Elective
Prerequisite	Operating Systems, Computer Networks	Antirequisite	None

Objectives
This course will introduce design concepts, frameworks, and applications in Edge Computing to the audience

Learning Outcomes:
Students are expected to:
1. Understand various edge devices and their ecosystems, issues and challenges
2. Develop edge-based distributed computing platforms and applications

Contents
Introduction of Edge and Fog Computing: Internet of Things (IoT) and New computing paradigms, Fog computing: A platform for Internet of Things and analytics, Emergence of edge computing, Legal aspects of operating IoT applications in the fog. (6 Lectures)
Edge Architecture: Multi-Tier cloud computing framework; Data services with clouds at home; Leveraging mobile devices to provide cloud service at the edge; Fast, scalable and secure onloading of edge functions. (8 Lectures)
Networking for Edge & Fog: Integrating IoT + Fog + Cloud Infrastructures: System modeling and research Challenges, Management and Orchestration of network slices in 5G, Fog, Edge, and Clouds. (6 Lectures)
System Design: Optimization problems in fog and edge computing, Middleware for fog and edge Computing: Design issues, A Lightweight container middleware for edge cloud architectures. (8 Lectures)
Data Processing: Data management in fog computing, Predictive analysis to support fog application deployment, Using machine learning for protecting the security and privacy of Internet of Things (IoT) systems, fog Computing realization for Big data analytics. (8 Lectures)
Applications and Case Studies: Fog computing realization for Big data analytics, exploiting fog computing in health monitoring, Smart surveillance video stream processing at the edge for real-time human objects tracking, Fog computing model for evolving smart transportation applications. (6 Lectures)

Text Book
R. BUYYA, S.N. SRIRAMA (2019), Fog and Edge Computing: Principles and Paradigms, Wiley-Blackwell, 2019.

Title	**Embedded Systems**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech, M.Tech., PhD	Type	Elective
Prerequisite	Computer Organization and Architecture	Antirequisite	None

Objectives

The Instructor will:

1. Explain the design of embedded systems and introduce concepts of different architectures and programming languages of embedded processors.

Learning Outcomes

The students are expected to have the ability to:

1. Program and to design embedded system using embedded processors
2. Design Embedded AI systems
3. Use different IDE and debugging tools

Contents

Introduction: Review of Embedded Computing, embedded system design process (4 lectures)

Architectures of embedded processors: Architecture of ARM Cortex M3, DSP and graphics processors,memory system mechanism, caches, memory management units and address translation, interfacing (10 lectures)

Programming and Software: models for program, data flow graphs, C and assembly language programming of ARM Cortex M3, Hardware- Software Co-design (12 lectures)

Embedded Operating Systems: Linux, Processes and real time operating systems; Multi-rate system; scheduling algorithms (8 lectures)

Embedded AI: Basics of embedded learning and adaptive systems, intelligent sensors, rule-based systems, hardware accelerators for AI, heterogeneous memory system design, current trends and future directions (8 lectures)

Textbook

1. Wolf, M., (2012), *Computers as Components: Principles of Embedded Computing System Design*, 3rd Edition, Elsevier.
2. Yiu, J., (2013), *The definitive Guide to ARM Cortex M3 and M4 Processors*, 3rd Edition, Elsevier.
3. Alippi, C., (2014), *Intelligence for Embedded Systems: A Methodological Approach*, Springer.

Preparatory Course Material

1. Mazidi, M.A., (2007), *The 8051 Microcontroller and Embedded Systems: Using Assembly and C,* 2nd Edition, Pearson Education India.

Title	**Environmental Informatics**	Number	CS L7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech (All), M.Tech, Ph.D.	Type	Elective
Prerequisite	Data Structure and Algorithms	Antirequisite	None

Objectives
The Instructor will:
1. Introduce techniques for environmental modeling, environmental databases and information systems.

Learning Outcomes
The students will have the ability to:
1. Apply the techniques for environment modeling on typical applications of environment monitoring and protection.
2. Appreciate the multidisciplinary nature of environmental problems.

Contents
Introduction to Environmental Informatics: environmental objects, characterisation of an environmental system, environmental metadata. [2 lectures]
Data capture and data storage: introduction, object taxonomies, mapping the environment, advanced techniques. [3 lectures]
Environmental sampling: sampling design, essentials for sampling of environmental data, satellite imagery [4 lectures]
Hydrological Modeling: Rainfall-Runoff models, Precipitation Models, Climate Models. [4 lectures]
Geographical information systems [4 lectures]
Environmental data analysis: Scales of operation of environmental data, re-sampling of environmental data, approximation of environmental data, trend estimation of environmental data [7 lectures]
Environmental statistics: probability distributions of environmental data, statistical measures, statistical tests, regression and correlation [6 lectures]
Environmental time series: fourier analysis, stationary processes, time correlation functions, frequency functions [7 lectures]
Environmental simulation models: modelling procedure, types and classification of mathematical models, process identification, an eutrophication simulator. [5 lectures]

Text Book
1. Günther, O., (1989). Environmental Information Systems. Springer, Berlin.
2. Box, G. E. P., Jenkins, G. M. & Reinsel, G. C., (1994). Time Series Analysis. 5th ed., Prentice Hall, Englewood Cliffs.

Reference Books
Avouris, N. M. and Page, B. (eds.), (1995). Environmental Informatics – Methodology and Applications of Environmental Information Processing. Kluwer, Dordrecht.

Course Title	**Ethics, Policy, Law and Regulations in AI**	Course No.	**CSL7xx**
Department	Computer Science and Engineering	Structure (L-T-P [C])	1-0-0 [1]
Offered for	B.Tech, MTech, PhD	Type	Elective
Prerequisite	Pattern Recognition and Machine Learning or ML or IML AI,	Antirequisite	None

Objectives

1. To understand implications and consequences of ethics, law, Regulations (data - privacy, ownership, data marketplace, etc) Policy in AI

Learning Outcomes

Students are expected to have the ability to develop an understanding of:

1. Ramifications of AI technologies on society
2. Ethical aspects of AI, ML and DS systems
3. Data privacy, ownership and IPR issues
4. Law, Regulations, Liabilities and Policies of AI, ML and DS systems

Contents

Ethics in AI
Law and Regulations
IPR
Policies
Case Studies

Text Book

1. Online resources
2. Will be added in future

Reference Books

Title	**Fundamentals of Machine Learning**	Number	CSL7670
Department	Computer Science and Engineering	L-T-P [C]	3–0–2 [4]
Offered for	Masters and PhD in Non-CS/AI Programs	Type	Elective
Prerequisite	Bridge Course of Programming (Python)	Antirequisite	None

Objectives
1. To introduce the fundamental theories of machine learning.
2. To equip the students with knowledge of implementation of data analysis and evaluation of machine learning algorithms.

Learning Outcomes
The students are expected to have the ability to:
1. Explain the basic concepts of ML driven data analysis
2. Utilize the tools and techniques for collecting, storing, securing, retrieving, and analyzing data of different modalities

Contents
Introduction to Probability, Bayes Theorem, Random Variable, Distribution Function and Probability Density (Mass) Function, Linear algebra. (8 lectures)

Evaluating ML Models: Precision, recall, specificity, sensitivity, predictive value, ROC curve, cross validation, Overfitting Underfitting, Bias and interpretability of ML models (4 lectures)

Unsupervised Learning, Feature Selection, Clustering, Missing data (6 Lectures)

Supervised Machine Learning Models: Regression Analysis, Bayes Classification, Parameter Estimation, Maximum Likelihood Estimator, k-nearest neighbor, Decision trees, Gradient Descent, Neural Networks, Ensemble Learning (16 lectures)

Introduction to Deep learning: CNN, Autoencoder, RNN (8 Lectures)

Textbook
1. Mitchell Tom (1997). Machine Learning, Tata McGraw-Hill
2. Wu, G., (2016), Machine Learning and Medical Imaging, Elsevier.

Reference
1. https://www.youtube.com/playlist?list=PLJ5C_6qdAvBGaabKHmVbtryZW9KpICiHC

Title	**GPU Programming**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech, M.Tech., PhD	Type	Elective
Prerequisite	Data Structures and Algorithms	Antirequisite	None

Objectives

The Instructor will:

1. Provide background to understand various aspects of Graphics Processing Unit (GPU)
2. Introduce parallel programming using GPUs.

Learning Outcomes

The students are expected to have the ability to:

1. Explain various concepts involving GPU Programming
2. Implement programs of GPU
3. Debug and profile parallel programs.

Contents

Introduction: History, graphics processors, graphics processing units, GPGPUs. Clock speeds, CPU / GPU comparisons, heterogeneity. Accelerators, parallel programming, CUDA / OpenCL / OpenACC (2 lectures)

Hello World Computation: Kernels, launch parameters, thread hierarchy, warps/wavefronts, thread blocks/workgroups, streaming multiprocessors, 1D / 2D / 3D thread mapping, device properties, simple programs (8 lectures)

Support: Debugging GPU programs. Profiling, profile tools, performance aspects (2 lectures)

Memory: Memory hierarchy, DRAM / global, local / shared, private/local, textures, constant memory, Pointers, parameter passing, arrays and dynamic memory, multi-dimensional arrays, Memory allocation, memory copying across devices, Programs with matrices, performance evaluation with different memories (5 lectures)

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 (6 lectures*)*

Functions: Device functions, host functions, kernels, functors, Using libraries (such as Thrust), developing libraries, (3 lectures)

Streams: Asynchronous processing, tasks, task-dependence, Overlapped data transfers, default stream, synchronization with streams, Events, event-based-synchronization - overlapping data transfer and kernel execution, pitfalls (6 lectures)

Advanced topics: Case studies, Dynamic Parallelism, Unified virtual memory, Multi-GPU processing, Peer access, Heterogeneous processing (8 lectures)

Textbook

1. Kirk,D. and Hwu,W., (2010), *Programming Massively Parallel Processors: A Hands-on Approach*, Hwu; Morgan Kaufman

Self-Learning Material

1. Cook,S., (2012), *CUDA Programming: A Developer's Guide to Parallel Computing with GPUs*, Morgan Kaufman

Preparatory Course Material

1. Introduction to Parallel Programming, https://www.youtube.com/watch?v=F620ommtjqk&list=PLAwxTw4SYaPnFKojVQrmyOGFCqHTxfdv2&index=1

Title	**Graph-theoretic Algorithms**	Number	**CS7xxx**
Department	Computer Science	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech, M. Tech, Ph.D.	Type	Elective
Prerequisite	Algorithm Design and Analysis	Antirequisite	None

Objectives
The Instructor will:
Introduce and address graph problems using algorithm design principles. Discuss graph theoretic algorithms in real-world scenarios.

Learning Outcomes
To give the student further exposure to the design, analysis, and application of algorithms for problems defined on graphs.
Formulate and solve real-world problems using the mathematical foundations of graph theory.

Contents
Introduction to graphs. (2 Lectures)
Max Flow Min Cut Theorem, Algorithms for min cut and max flow, Maximum matching, Hall's theorem, algorithms for computing maximum matching in weighted and unweighted graphs. (6 Lectures)
Vertex and Edge Coloring of Graphs, Independent Set. (4 Lectures)
Interval graphs, Perfect elimination orders, Comparability graphs, Interval Graph Recognition, Friends of Interval Graphs. (7 Lectures)
Trees and treewidth. (7 Lectures)
Planar Graphs, Problems in planar graphs, Maximum Cut. (8 Lectures)
Planarity Testing, Triangulated graphs, Friends of planar graphs, Hereditary properties. (8 Lectures)

Textbooks
A. BRANDSTÄDT, V.B. LE, J.P. SPINRAD (1999), Graph Classes: A Survey, SIAM Monographs on Discrete Mathematics and Applications, SIAM.
T. NISHIZEKI, N. CHIBA (1988), Planar Graphs: Theory and Algorithms, North-Holland publisher.
M.C. GOLUMBIC (2004), Algorithmic Graph Theory and Perfect Graphs, Academic Press, 2nd Edition.

Self-Learning Material
T. BIEDL, Lecture Notes of a Graduate Course, University of Waterloo.

Title	**Graph Theory and Applications**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech, M.Tech., Ph.D.	Type	Elective
Prerequisite	None	Antirequisite	None

Objectives

The Instructor will:

1. Introduce various terminologies, concepts and algorithms related to graphs, and discuss their applications in real-world scenarios.

Learning Outcomes

The students are expected to have the ability to:

1. Formulate and solve real-world problems using the mathematical foundations of graph theory.

Contents

Preliminaries: Graphs, Isomorphism, Subgraphs, Matrix representations, Degree, Operations on graphs, Degree sequences (3 lectures)
Connected graphs and shortest paths: Walks, Trails, Paths, Connected graphs, Distance, Cut-vertices, Cut-edges, Blocks, Connectivity, Weighted graphs, Shortest path algorithms (4 lectures)
Trees: Characterizations, Number of trees, Minimum spanning trees (3 lectures)
Special classes of graphs: Bipartite graphs, Line graphs, Chordal graphs (2 lectures)
Eulerian graphs: Characterization, Fleury's algorithm, Chinese-postman-problem (2 lectures)
Hamilton graphs: Necessary conditions and sufficient conditions (3 lectures)
Independent sets, coverings, matchings: Basic equations, Matchings in bipartite graphs, Perfect matchings, Greedy and approximation algorithms (6 lectures)
Vertex colorings: Chromatic number and cliques, Greedy coloring algorithm, Coloring of chordal graphs, Brook's theorem (2 lectures)
Edge colorings: Gupta-Vizing theorem, Class-1 graphs and Class-2 graphs, Equitable edge-coloring (5 lectures)
Planar graphs: Basic concepts, Euler's formula, Polyhedrons and planar graphs, Characterizations, Planarity testing, 5-color-theorem (3 lectures)
Directed graphs: Out-degree, In-degree, Connectivity, Orientation, Eulerian directed graphs, Hamilton directed graphs (5 lectures)
Applications: Biology, Social Sciences, Engineering, Computer Science (4 lectures)

Textbook

1. West,D.B., (2002), *Introduction to Graph Theory*, 2nd Edition, Prentice Hall of India
2. Deo,N., (2003), *Graph Theory: With Application to Engineering and Computer Science*, Prentice Hall of India

Reference Books

1. Research literature

Self Learning Material

1. NPTEL: Graph Theory (for CSE), https://nptel.ac.in/courses/106108054/39
2. NPTEL: Graph Theory (for Mathematics), https://nptel.ac.in/courses/111106050/

Title	**Hardware Software Co-design**	Number	EEL7XX/CS7XX
Department	CSE/Electrical Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech.	Type	Elective
Prerequisite	Digital Design	Antirequisite	None

Objectives
The Instructor will:
Give overview of the cooperative design of hardware and software components.
Provide the understanding of unification of currently separate hardware and software paths

Learning Outcomes
The students are expected to have the ability to:
Achieve system-level objectives by exploiting the synergism of hardware and software through their concurrent design.
Understand the movement of functionality between hardware and software.

Contents
Introduction to Hardware-Software Co-design, Design process, Application domains, Design technologies. (5 Lectures)

Data flow modeling implementation in software and hardware, Analysis of control flow and data flow, Finite state machine with datapath, Custom hardware, Microprogrammed architecture, system on chip. (12 Lectures)

Codesign methodologies: Model representation, Hardware-software mapping - partitioning, scheduling, and allocation, Software code optimization. (10 Lectures)

Hardware-software interfaces: Principles of hardware software communication, On-chip buses, Memory- mapped interface, Coprocessor interface. (10 Lectures)

Codesign CAD tools: Ptolemy Project. (5 Lectures)

Textbook
P.R. SCHAUMONT (2010), A Practical Introduction to Hardware/Software Codesign, Springer; 1st Edition.
G. De MICHELI, M. SAMI (1996), Hardware/Software Co-Design, Springer.

Title	**Health Informatics**	Number	CS L7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech (All), M.Tech, Ph.D.	Type	Elective
Prerequisite	Data Structure and Algorithms	Antirequisite	None

Objectives
1. Develop knowledge about problems and challenges that health informatics addresses
2. To develop an understanding of different types of data exchange formats and standards techniques for different purposes.
3. To equip the students with various possible applications of digital data analysis in healthcare.

Learning Outcomes
The students are expected to have the ability to:
1. Apply basic knowledge to the research and practice of health informatics
2. Demonstrate basic skills and knowledge in health informatics for application in future health-related careers
3. Utilize the tools and techniques for collecting, storing, securing, retrieving, and reporting health care data and information

Contents
Introduction: Overview, health data, information, and knowledge, electronic health records, system architecture for EHRs, ambulatory EHR functions and interoperability, personal health records and decision aids. *[6 lectures]*
Health Care Information Systems: Healthcare data standards - Overview, methods, protocols, terminologies, and specifications for the collection, exchange, storage, and retrieval of information associated with health care applications. *[6 lectures]*
Data Collection and Quality Assurance: Data collection, cleaning data, managing change, Using Data for Care Delivery, Coordination, and Quality Improvement, Quality assurance, Security, Privacy. *[8 lectures]*
Clinical Data: Introduction, medical digital data formats and exchange, medical image formats such as DICOM in the context of health informatics. *[4 lectures]*
Informatics: Information retrieval, bioinformatics - usage of informatics in genomics and other aspects of molecular biology, patient case history analytics, applications in public health, ethical issue in health informatics. *[8 lectures]*
Medical Diagnostic Decision Support: probabilistic approaches, clinical scores, logical approaches - expert system, inference engines, case studies. *[6 Lectures]*
Digital healthcare: Telemedicine, mobile for health, public health *[4 lectures]*

Text Books
1. Hoyt, R. E., & Yoshihashi, A. K. (2014). Health informatics: practical guide for healthcare and information technology professionals. Lulu. com
2. Hersh, W. R., & Hoyt, R. E. (2018). Health Informatics: Practical Guide Seventh Edition. Lulu. com

Reference Material
1. Landais, P., Boudemaghe, T., Suehs, C., Dedet, G., Lebihan-Benjamin, C., Venot, A., ... & Quantin, C. (2014). Medical Informatics, e-Health: Fundamentals and Applications.
2. Healthcare data standards, https://www.ncbi.nlm.nih.gov/books/NBK216088/
3. Data Collection and Quality Assurance, https://www.ncbi.nlm.nih.gov/books/NBK208601/
4. Computer-aided diagnosis medical image analysis techniques, https://www.intechopen.com/books/breast-imaging/computer-aided-diagnosis-medical-image- analysis-techniques

Self-Learning Material
Health Informatics Specialization, Coursera Online, https://www.coursera.org/specializations/health-informatics

Course Title	**High-level Synthesis**	Course No.	**CS/EE-7XX**
Department	CSE, EE	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B.Tech, M.Tech, Ph.D	Type	Elective
Prerequisite	Basic knowledge of digital circuits	Antirequisite	

Objectives
To provide an introduction to high-level synthesis of digital circuits and electronic design automation. To explain the HLS design flow: circuit, logic, RTL, algorithm, systems.

Learning Outcomes
The students are expected to have the ability in:
Design Digital Circuits, Logic synthesis, optimization and technology mapping. Design Verification
Use different tools like C-to-Silicon, AutoPilot from AutoESL, Synopsis for logic synthesis,

Contents
Introduction: Basic overview of the VLSI design flow, design abstractions (2 Lectures)
Digital Sub-System Design: Logic Circuits, Adders, Multipliers, ALU (5 Lectures)
HDL language: System C (8 Lectures)
Synthesis: Scheduling, Allocation, Binding (3 Lectures)
RTL Synthesis, Retiming, FSM Encoding, Combinational logic optimization and technology mapping *(6 Lectures)*
Verification: Design vs verification, Types of Verification at each step of ASIC Design Flow (functional/power/Gate Level Simulation (GLS)), UVM, Verification planning and Designing a test bench (7 Lectures)
Chip Architectures: Full Custom, Standard Cells and FPGA (4 Lectures)
Synthesis Tool: C-to-Silicon, AutoPilot from AutoESL (4 Lectures)
Overview of layout synthesis topics: Placement and routing. Chip Synthesis. (3 Lectures)

Reference Books
1. M. FINGEROFF (2010), High-level Synthesis Blue Book, Xlibris.
2. G. De MICHELI (2017), Synthesis and Optimization of Digital Circuits, McGraw-Hill.
3. S. KANG, Y. LEBLEBICI (2003), MOS Digital Integrated Circuits, Analysis and Design, WCB- MCGrawHill, 3rd Edition.

Course Title	**Information Retrieval**	Course No.	**CSL7XXX**
Department	Computer Science and Engineering	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B.Tech, M. Tech, Ph.D.	Type	Elective
Prerequisite	PRML / Introduction to ML	Antirequisite	None

Objectives

Information retrieval covers the tasks of indexing, searching, and recalling data, particularly text or other unstructured forms. It has an important role to play in a large number of applications viz., digital libraries, office automation, internet and e-commerce. The aim of the course is to study theoretical aspects as well as implementation issues of classical and modern retrieval problems.

Learning Outcomes

The students are expected to learn and gather expertise in:

1. The underlying technologies of modern information retrieval system
2. Developing new search engines with high search accuracy

Contents

Introduction to Information Retrieval: The nature of unstructured and semi-structured text, Inverted index and Boolean queries. (2 Lectures)

Search Engine Architecture: Basic building blocks of a modern search engine system: web crawler, basic text analysis techniques, Inverted index, Query processing, Search result interface, Semantic search using Ontology (7 Lectures)

Retrieval Models: Boolean vector space, TFIDF, Okapi, Probabilistic language modeling, Latent semantic indexing, Vector space scoring, The cosine measure, Efficiency considerations, Document length normalization, Relevance feedback and query expansion, Rocchio, Ontological models (8 Lectures)

Performance Evaluation: Evaluating search engines, User happiness, Precision, Recall, F-measure, Creating Test collections: kappa measure, Interjudge agreement. (5 Lectures)

Text Categorization and Filtering: Introduction to text classification, Naive Bayes model, Spam filtering, Vector space classification using hyperplanes, Centroids, K-Nearest Neighbors, Support vector machine classifiers, Kernel functions, Boosting. (6 Lectures)

Text Clustering: Clustering versus classification, Partitioning methods, K-means clustering, Gaussian mixture model, Hierarchical agglomerative clustering, Clustering terms using documents. (6 Lectures)

Advanced Topics: Summarization, Topic detection and tracking, Personalization, Question answering, Cross language information retrieval (3 Lectures)

Web Information Retrieval: Hypertext, web crawling, Search engines, Ranking, Link analysis, PageRank, HITS. (3 Lectures)

Retrieving Structured Documents: XML retrieval, Semantic web (2 Lectures)

Textbooks

1. C.D. MANNING, P. RAGHAVAN, H. SCHUETZE (2008), Introduction to Information Retrieval, Cambridge University Press.
2. B. CROFT, D. METZLER, T. STROHMAN (2010), Search Engines: Information Retrieval in Practice, Pearson Education.
3. B. RICARDO, B. RIBEIRO-NETO (2011), Modern Information Retrieval, Addison-Wesley, 2nd Edition.

Title	**Introduction to AR and VR**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B. Tech., M.Tech., PhD	Type	Elective
Prerequisite	Computer Graphics	Antirequisite	None

Objectives
The Instructor will:
1. Discusses such issues, focusing upon the human element of AR and VR
2. Explain the Hardware and software related issues related to AR and VR

Learning Outcomes
The students are expected to have the ability to:
1. Explain perceptual concepts governing augmented reality and virtual reality
2. Identify and solve the issues of various augmented reality and virtual reality frameworks
3. Design immersive experience using AR and VR Software

Contents
(Fractal 1)
Introduction: Definition of X-R (AR, VR, MR), modern experiences, historical perspective, Hardware, sensors, displays, software, virtual world generator, game engines (6 Lectures)
Geometry of Visual World: Geometric modeling, transforming rigid bodies, yaw, pitch, roll, axis-angle representation, quaternions, 3D rotation inverses and conversions, homogeneous transforms, transforms to displays, look-at, and eye transform, canonical view and perspective transform, viewport transforms (8 Lectures)

(Fractal 2)
Light and Optics: Interpretation of light, reflection, optical systems (4 Lectures)
Visual Perception: Photoreceptors, Eye and Vision, Motion, Depth Perception, Frame rates and displays (6 Lectures)
Tracking: Orientation, Tilt, Drift, Yaw, Lighthouse approach (4 Lectures)

(Fractal 3)
Head Mounted Display: Optics, Inertial Measurement Units, Orientation Tracking with IMUs, Panoramic Imaging and Cinematic VR, Audio (8 Lectures)
Frontiers: Touch, haptics, taste, smell, robotic interfaces, telepresence, brain-machine interfaces (6 Lectures)

Textbook

1. Shirley, M., (2016), *Fundamentals of Computer Graphics*, 4th Edition, CRC Press
2. LaValle, (2016), *Virtual Reality*, Cambridge University Press
3. Schmalstieg D, and Hollerer T. (2016). *Augmented Reality: Principles & Practice*, Pearson Education India

Reference Books

1. Jerald,J., (2015), *The VR Book: Human-Centered Design for Virtual Reality*, Morgan & Claypool
2. Mather,G., (2009), *Foundations of Sensation and Perception*, 2nd Edition, Psychology Press
3. Shirley,P., Ashikhmin,M., Marschner,S. and Peters,A.K., *Fundamentals of Computer Graphics*, 3rd Edition, CRC Press
4. Bowman,D.A., Kruijff,E., LaViola,J.J. and Poupyrev,I., (2014), *3D User Interfaces: Theory and Practice*, 2nd Edition, Addison Wesley Professional

Self Learning Material
1. Steven M. LaValle, Video Lectures, https://www.youtube.com/playlist?list=PLbMVogVj5nJSyt80VRXYC-YrAvQuUb6dh

Title	**Introduction to Blockchain**	Course No.	CSL 7XX
Department	Computer Science and Engineering	Structure (L-T-P [C])	3–0–0 [3]
Offered for	B.Tech, MTech, PhD.	Type	Elective
Prerequisite	Cryptography / Cryptography and Network Security	Antirequisite	Blockchain

Objectives
The Instructor will:
Explain how blockchain technology works
Integrate blockchain technology into the current business processes to make them secure

Learning Outcomes
The students are expected to have the ability to:
Understand what and why of Blockchain
Explore major components of Blockchain and Identify a use case for a
Blockchain application Create their own Blockchain network application

Contents
Introduction to Blockchain: Digital Trust, Asset, Transactions, Distributed Ledger Technology, Types of network, Components of blockchain (cryptography, ledgers, consensus, smart contracts). (5 Lectures)
PKI and Cryptography: Private keys, Public keys, Hashing, Digital Signature. (6 Lectures)
Consensus: Byzantine Fault, Proof of Work, Proof of Stake. (6 Lectures)
Cryptocurrency: Bitcoin creation and economy, Limited Supply and Deflation, Hacks, Ethereum concept and Ethereum classic. (10 Lectures)
Hyperledger Fabric: Hyperledger Architecture, Membership, Blockchain, Transaction, Chaincode, Hyperledger
Fabric, Features of Hyperledger, Fabric Demo. (8 Lectures)
Blockchain Applications: Building on the Blockchain, Ethereum Interaction - Smart Contract and Token (Fungible, non-fungible), Languages, Blockchain-as-a-service. (6 Lectures)

Textbook
A. BAHGA, V. MADISETTI (2017), Blockchain Applications: A Hands-On Approach, VPT.

Self Learning Material
1. M. SWAN (2015), Blockchain: Blueprint for a New Economy, O'Reilly Media.
2. R. WATTENHOFER (2016), The Science of the Blockchain, CreateSpace Independent Publishing Platform.
3. I. BASHIR (2017), Mastering blockchain, Packt Publishing Ltd.
4. K.E. LEVY, Book-smart, Not Street-smart: Blockchain-based Smart Contracts and the Social Workings of Law, Engaging Science, Technology, and Society, vol. 3, pp. 1-15, 2017.

Preparatory Course Material
MIT Online Blockchain Course, Learn Blockchain Technology: https://getsmarter.mit.edu/

Course Title	**Introduction to Wireless Ad hoc Networks**	Course No.	**CS7XX**
Department	Computer Science and Engineering	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B.Tech, MTech, PhD	Type	Elective
Pre-requisites	Computer Networks	Antirequisite	None

Objectives
The course is intended to provide students with an understanding of the wide applicability of Wireless Ad-Hoc networks and their associated design issues and challenges.
It also sheds light on protocol design, mainly at the MAC layer.

Learning Outcomes
At the end of this course, students would be able to appreciate the mechanisms and interventions required at different levels to make Wireless Ad-hoc networks work in practice.
They would be able to explain different MAC protocols and their issues.
Students would also be able to contrast transport layer design issues arising out of mobility, with the conventional TCP.

Contents
Introduction to Wireless Ad-hoc Networks: Overview about Mobile ad-hoc networks (MANETs), Vehicular Ad- hoc networks (VANETs), Wireless Sensor Networks (WSNs) and Wireless Mesh Networks (WMNs). (2 Lectures)
Physical layer: Modulation techniques, Channel models, Case-study of 802.11a PHY. (3 Lectures)
Link Layer: Single-hop MAC protocols, multi-hop MAC protocols, Error correcting codes. (5 Lectures)
Network Layer: Mobile IP, Distributed wireless routing algorithms (AODV, DSDV, DSR, OLSR), Routing metrics. (8 Lectures)
Transport Layer: TCP over wireless, Transport level mobility management, Multihop transport protocols. (6 Lectures)
Introduction to 5G networks (3 Lectures)
Low Power Wide Area Networks (LPWAN) for IoT: LoraWAN, NB-IoT, SigFox. (9 Lectures)
Personal Area Networks: Bluetooth. (3 Lectures)
Future trends in wireless networks: 802.11ax, 802.11ay, mm-Wave WiFi. (3 Lectures)

Note
Please note that the course material would not be taken from a single book or resource. It would be a combination of textbook material, research papers and other sources.

Textbooks
W. STALLINGS (2009), Wireless Communications & Networks, Pearson Education India, 2nd Edition.
T.S. RAPPAPORT (2010), Wireless Communications: Principles and Practice, Prentice Hall, 2nd Edition.

Reference Books
C.S.R. MURTHY, B.S. MANOJ (2004), Ad hoc Wireless Networks: Architectures and Protocols, Pearson.
G. AGGELOU (2004), Mobile ad hoc networks: from wireless LANs to 4G networks, McGraw-Hill Professional.
S. BASAGNI, M. CONTI, S. GIORDANO, I. STOJMENOVIC, (Eds.), Mobile Ad hoc Networking, John Wiley, 2004.
A. GOLDSMITH (2005), Wireless Communications, Cambridge university press.
Recent relevant RFCs, Internet drafts, selected research papers from relevant venues: Mobicom, MobiSys, SIGCOMM, Infocom, IEEE TMC, ACM MC2R.

Self-learning Material
S. MISHRA NPTEL, IIT Kharagpur: https://nptel.ac.in/courses/106105160/
Relevant research papers.

Title	**Machine Learning in Epidemiology**	Number	CS L7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech (All), M.Tech, Ph.D.	Type	Elective
Prerequisite	Machine Learning / PRML/IML	Antirequisite	None

Objectives
Course focuses on advances in machine learning and its application to causal inference and prediction via *Targeted Learning*, which allows the use of machine learning algorithms for prediction and estimating so-called causal parameters, such as average treatment effects, optimal treatment regimes, etc.

Learning Outcomes
On completion of the course the students will have:
1. Ability to apply machine learning algorithms to prediction problems and estimate and derive inference for the resulting fit.
2. Ability to use the fits of machine learning algorithms to estimate causal effects using simple substitution estimators.
3. Knowledge of How the general methodology applies to goals of Precision Medicine.

Contents
Introduction*:* Introduction to epidemiology, statistical methodologies in epidemiologic studies, curse of dimensionality in epidemiology, Relation between ML and epidemiology, more on epidemiology. *[4 Lectures]*
Prediction of Observational Epidemiology*:* Clinical Decision Support, Disease Surveillance, Introduction to Targeted Machine Learning. *[4 Lectures]*
Causal Inference: Understanding bias, Correlation vs. causality, Causal graph inference. *[4 Lectures]*
Super-Learning: Parametric Models, Data Adaptivity, Performance Measure on Independent Data, Customizable Optimality criterion, Sequential super learning. *[8 Lectures]*
Target Update of Machine Learning: TMLE, HAL, HAL-TMLE, C-TMLE, TMLE with Clustering, On- line super learning, On-line targeted learning for time series. *[14 Lectures]*
Case Studies in Epidemiology*:* Ebola, Corona etc. *[4 Lectures]*
Future Directions: Deep learning in clinical epidemiology, Recurrent unit neural networks to identify individuals at risk. *[4 Lectures]*

Reference Books
1. Van der Laan & Rose (2018), Targeted Learning in Data Science Causal Inference for Complex Longitudinal Studies, Springer
2. Van der Laan & Rose (2011), Targeted Learning in Causal Inference for Observational and Experimental Data, Springer

Reference Articles
1. Timothy L. Wiemken, Robert R. Kelley (2020), Machine Learning in Epidemiology and Health Outcomes Research, *Annual Review of Public Health*, 41(1), 21-36
2. Goldstein, N. D., LeVasseur, M., & McClure, L. A. (2020). On the Convergence of Epidemiology, Biostatistics, and Data Science. *Harvard Data Science Review.*
3. Char DS, Shah NH, Magnus D. (2018), Implementing machine learning in health care- Addressing ethical challenges, *New England Journal of Medicine*; 378-981 (3).
4. Chiavegatto Filho ADP, Dos Santos HG, do Nascimento CF, Massa K, Kawachi I. (2018), Overachieving municipalities in public health: A machine learning approach, Epidemiology; 29-836- 40.
5. https://medium.com/causal-data-science/causal-data-science-721ed63a4027
6. Other relevant articles

Title	**Machine Learning with Big Data**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech, M.Tech., PhD	Type	Elective
Prerequisite	Machine Learning / PRML	Antirequisite	None

Objectives

The Instructor will:

1. Provide an understanding of the role of big data in the real-world scenarios
2. Provide technical details about various algorithms and software/hardware tools/platforms related to big data

Learning Outcomes

Students are expected to have the ability to:

1. Develop an understanding of big data in the modern context, and independently work on problems relating to big-data
2. Design and program efficient algorithms for big data from the perspective of a project

Contents

Introduction: What is big data, Unreasonable effectiveness of data (1 lecture)
Streaming algorithms: Streaming Naive Bayes, Stream and sort (2 lectures)
Platforms for learning from big data: MapReduce, New Software Stack, Large Scale File System Organization (5 lectures)
Nearest Neighbour Search, Jaccardi Similarity of Sets, Similarity of Documents, Locality Sensitive Hashing, The Stream Data Model (4 lectures)
Randomized methods: Clustering, Hashing, Sketching, Scalable stochastic gradient descent (3 lectures)
Frequent Itemsets: The Market Basket Model, A-Priori Algorithm, Handling larger datasets in Main Memory, Limited-Pass Algorithms, Counting Frequent Items in a Stream (6 lectures)
Parameter Servers: Introduction, Abstraction, Parameter Cache Synchronization, Asynchronous execution, Model Parallel Examples (3 lectures)
Graph-based methods (6 lectures)
Page Rank, Topic Sensitive Page Rank, Approaches to Page Rank iteration, Link Spam, Semi-supervised learning, Scalable link analysis, Models for Recommendation Systems, Social Networks as Graphs (9 lectures)
Large-scale Machine Learning with CPUs and GPUs (3 lectures)

Textbook

1. Leskovec,J., Rajaraman,A., Ullman,J., (2014), *Mining of Massive Datasets*, 2nd Edition, Cambridge University Press
2. Bekkerman,R., Bilenko,M., Langford,J., (2011), *Scaling Up Machine Learning*, Cambridge University Press

Reference Books

1. Research literature

Self Learning Material

1. Department of Machine Learning, Carnegie Mellon University, Machine Learning with Large Datasets Course
2. Department of Computer Science, University of California, Berkeley, Scalable Machine Learning
3. ETH Zurich, Data Mining: Machine Learning from Large Datasets

Title	**Medical Image Analysis**	Number	CSL7xx
Department	CSE, AI&DS, EE	L-T-P [C]	3–0–0 [3]
Offered for	BTech, MTech, Ph.D.	Type	Elective
Prerequisite	Digital Image Processing / Computer Vision, Machine Learning/PRML	Antirequisite	None

Objectives
The Instructor will provide an in-depth understanding of classical and machine learning based techniques for medical image analysis.

Learning Outcomes
The students will have ability to:
1. interpret and analyse the images in a quantitative way.
2. apply the learned techniques for novel disease diagnosis and prognosis.

Contents
Classical Approaches: *[1-0-0]*
Introduction to image processing and medical imaging modalities, denoising and enhancement [4 Lectures] Tissue and Cell Segmentation: clustering, active contours and level sets based approaches [5 Lectures] Medical Image alignment: rigid and deformable registration [5 Lectures]

Machine Learning and Deep Learning Approaches*: [2-0-0]*
Fundus Image analysis, Retinal Vessel Segmentation [4 Lectures]
MRI image analysis and segmentation, 3D brain reconstruction from and MRI slices and analysis [5 Lectures]
Microscopic image analysis and interpretation [5 Lectures]
Ultrasonography image analysis [4 Lectures]
X-Ray and CT image segmentation, diagnosis and prognosis of various diseases [5 Lectures]
Correlation between different medical imaging modalities and conversions, augmenting clinical measurements with medical imaging modalities for diseases diagnosis and prognosis [5 Lectures]

Text Books
1. Prince, J. L., & Links, J. M. (2006). *Medical imaging signals and systems*. Upper Saddle River, NJ: Pearson Prentice Hall.
2. Suetens, P. (2017). Fundamentals of medical imaging. Cambridge university press.

Self-Learning Material:
Medical Image Processing by Prof. Jeff Orchard at University of Waterloo:
https://cs.uwaterloo.ca/~jorchard/cs473/CS473/Welcome.html

Course Title	**Mobile and Pervasive Computing**	Course No.	**CS7XX**
Department	Computer Science and Engineering	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B. Tech and M. Tech	Type	Elective
Pre-requisites	Computer Networks	Antirequisite	None

Objectives
1. Explain necessary concepts to understand challenges in developing applications for mobile devices, when contrasted with that of a general-purpose computer.
2. Introduce mechanisms employed by different cellular technologies, particularly 5G, and contrast them with each other.

Learning Outcomes
At the end of this course, students would be able to:
1. Describe the working principles of various cellular technologies, as well as appreciate the associated challenges.
2. Compare/contrast various aspects of Mobile and Pervasive Computing, viz. localization, sensing, security and privacy concerns, application development, etc.
3. Implement the theoretical concepts into practical scenarios..

Contents
Overview of Wireless Systems: Infrastructure-based vs Ad-hoc, Wireless LANs, Cellular systems, Sensor networks, Bluetooth, WiFi, WiMAX. (3 Lectures)
Medium Access: Link Adaptation, Routing Protocols. (4 Lectures)
***Mobility and Handoff Management**:* Link layer mobility mechanisms (location management protocols), Network layer mobility mechanisms (Macro and Micro mobility protocols), Handoff management protocols. (6 Lectures)
Cellular Networks: LTE and 5G overview, 5G Architecture, RAN and dynamic CRAN, Mobility management and Network slicing in 5G. (11 Lectures)
Pervasive Computing: Principles, Characteristics, Pervasive devices, Smart sensors and actuators, Context communication and access services. (4 lectures)
Context Aware Sensor Networks: Open protocols (SDP, Jini, SLP, UPnP), SyncML framework, Context aware mobile services, Context aware security. (4 lectures)
Energy Efficiency, Localization **(GPS/WiFi/GSM),** ***Security*** (4 Lectures)
Recent Advances in Wearable devices (3 Lectures)
Body Area Networks **(BAN)** (3 Lectures)

Textbooks
1. I. STOJMENOVIC (2002), Handbook of Wireless Networks and Mobile and Pervasive Computing, Wiley.
2. S. LOKE (2006), Context-aware Pervasive Systems: Architectures for a New Breed of Applications, CRC Press.
3. A. OSSEIRAN, J.F. MONSERRAT, P. MARSCH, (Eds.), 5G Mobile and Wireless Communications Technology, Cambridge University Press, 2016.

Reference Books
1. R. KAMAL (2008), Mobile and Pervasive Computing, Oxford University Press, 3rd Edition.
2. L. MERK, M. NICLOUS (2006), Principles of Mobile and Pervasive Computing, Dreamtech Press, 2nd Edition.
3. G. AGGELOU (2004), Mobile Ad hoc Networks: From Wireless LANs to 4G Networks, McGraw-Hill Professional.

Self-learning Material:
1. J.J. DRAKE, Z. LANIER, C. MULLINER, P.O. FORA, S.A. RIDLEY, G. WICHERSKI (2014), Android Hacker's Handbook, John Wiley, 1st Edition.
2. P. SINGH, NPTEL, IIIT Delhi: https://nptel.ac.in/courses/106106147/
3. Relevant research papers from MobiSys, Sensys, MC2R, MobiCom and UbiComp.

Course Title	**5G Mobile Networks**	Course No.	**CS7XX**
Department	Computer Science and Engineering	Structure (L-T-P-C)	3-0-0
Offered for	B.Tech, M.Tech, PhD	Type	Elective
Pre-requisites	Computer Networks, Introduction to Wireless Ad hoc Networks	Antirequisite	None

Objectives
The course is intended to provide students with an understanding of the forthcoming 5G technology, and its ability to transform the world for the better.

Learning Outcomes
Students would be able to appreciate the increasing role that SDNs play in 5G systems and beyond.

Contents
4G LTE networks: Introduction, overview and architecture. (4 Lectures)
From 4G to 5G : Why, When and How? (1 Lecture)
5G overview: Understanding 5GPP & NGMN, 5G architecture and design objective, 5G spectrum requirements, SDN in the 5G context. (6 Lectures)
5G Core: 5G RAN & Dynamic CRAN, Virtual Network Functions in the 5G core, 5G NR Logical Architecture, 5G Protocol Stack. (13 Lectures)
5G signalling: Millimeter wave propagation, Distributed massive MIMO principle, 5G Ultra dense cellular networks, 5G Coordinated Multi-Point. (8 Lectures)
5G Mobile Edge computing & Fog computing (2 Lectures)
5G applications: Healthcare, Transportation, Smart cities, AR, VR. (2 Lectures)
5G vs mm-wave Wifi: A comparison. (2 Lectures)
India's 5G policy and vision (1 Lecture)
The 6G vision (1 Lecture)
Paper discussions (2 Lectures)

Note
Please note that the course material would not be based on a single book or resource. It would be a combination of textbook material, research papers and other sources from the Internet.

Textbooks
A. OSSEIRAN, J.F. MONSERRAT, P. MARSCH (Eds.) (2016), 5G Mobile and Wireless Communications Technology, Cambridge University Press.

Reference Books
A. ELNASHAR, M.A. EL-SAIDNY, M. SHERIF (2014), Design, Deployment and Performance of 4G-LTE Networks: A Practical Approach, John Wiley & Sons.

Self-learning Material
Relevant research papers.

Title	**Natural Language Understanding**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech, M.Tech., PhD	Type	Elective
Prerequisite	Deep Learning	Antirequisite	None

Objectives

The Instructor will:

1. To provide insights into fundamental concepts and algorithms related to Natural Language Understanding
2. Impart working expertise by introducing practical problems.

Learning Outcomes

The students are expected to have the ability to:

1. Formulate natural language understanding tasks
2. Design and implement basic applications of NLU

Contents

Traditional NLU: Introduction to NLU, Motivation, Morphology, Parts-of-Speech, Language Models, Word Sense Disambiguation, Anaphora Resolution, Basics of Supervised and Semi-supervised Learning for NLU, Hidden Markov Models for language modeling, EM Algorithm, Structured Prediction, Dependency Parsing, Topic Models, Semantic Parsing, Sentiment analysis. (14 Lectures)

Deep Learning for NLU: Intro to Neural NLU, Word Vector representations, Neural Networks and backpropagation -- for named entity recognition, Practical tips: gradient checks, overfitting, regularization, activation functions, Recurrent neural networks -- for language modeling and other tasks, GRUs and LSTMs -- for machine translation, Recursive neural networks -- for parsing, Convolutional neural networks -- for sentence classification, Question answering and dialogue system, Graph Neural Network for NLU, Natural Language Generation, Analysis and Interpretability of Neural NLU. (22 Lectures)

Knowledge Graphs: Knowledge graph embedding techniques, Inference on knowledge graphs. (6 Lectures)

Textbook

1. C. MANNING, H. SCHUTZE (1999), Foundations of Statistical Natural Language Processing, MIT Press.
2. D. JURAFSKY, J.H. MARTIN, Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition (3rd Edition Draft), 2019.

Reference Books:

1. E. BENDER (2013), Linguistic Fundamentals for NLP, Morgan Claypool Publishers..
2. J. ALLEN (1995), Natural Language Understanding, Pearson Education, 1995.
3. Research Literature.

Self-Learning Material

1. http://web.stanford.edu/class/cs224n/index.html#schedule (Deep learning for NLP)

Title	**Neuromorphic Computing and Design**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech, M.Tech., PhD	Type	Elective
Prerequisite	Artificial Intelligence, PRML/Introduction to ML	Antirequisite	None

Objectives

The Instructor will:

1. Provide information about neuroscientific progress towards reverse-engineering the brain
2. Provide essentials on key hardware building blocks, system level VLSI design and practical real-world applications of neuromorphic Systems

Learning Outcomes

The students are expected to have the ability to:

1. View neuromorphic computing as a computer architecture research problem
2. Perform software and hardware implementation of basic biological neural circuits

Contents

CSL7xx1 Introduction to Neuromorphic Engineering 1-0-0 [1]

Foundational Concepts: Introduction to neuromorphic engineering, neuroanatomy of human brain, signaling and operation of biological neurons, neuron models - LIF, IF, HH, synapses and plasticity rules, spike-time-dependent plasticity (STDP), biological neural circuits, non-von Neumann computing approach, learning rules, retina, cochlea (14 lectures)

CSL7xx2 Neuromorphic Computing 1-0-0 [1]

Neuromorphic Computing: Spiking Neural Networks (SNN), Advanced Nanodevices for Neuron Implementation, Synaptic emulation - non-volatile memory (NVM), Flash, RRAM, memristors, CNT, Case study on Intel's Loihi neuromorphic chip (14 lectures)

CSL7xx3 Neuromorphic Hardware Implementation 1-0-0 [1]

Hardware Implementation: Electronic synapses, Digital/Analog neuromorphic VLSI, Hardware Implementation of Neuron circuits, Hardware Implementation of Synaptic and Learning circuits, Synaptic programming methodology optimization (14 lectures)

Textbook

1. Liu, S.C., (2002), *Analog VLSI: Circuits and Principles*, MIT Press.
2. Kozma, R., (2012), *Advances in Neuromorphic Memristor Science*, Springer.
3. Kandel, E., (2012), *Principles of neural science*, McGraw Hill.

Title	**Parameterized Complexity**	Number	**CS7xxx**
Department	Computer Science	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech, M. Tech, Ph.D.	Type	Elective
Prerequisite	Algorithm Design and Analysis	Antirequisite	None

Objectives
The Instructor will:
The objective of the course is to introduce a technique to deal with NP-hard problems

Learning Outcomes
Learn a new set of techniques to cope with NP-hard problems. Identify novel and significant open research questions in the field.

Contents
Introduction to Parameterized Complexity. (1 Lecture)
Basic Techniques to design parameterized algorithms: Bounded Search Tree, Iterative Compression, Randomized method. (16 Lectures)
kernelization: Greedy Based Kernels, Matching Based Kernels, Sunflower Lemma, Crown Decomposition, Expansion Lemma. (16 Lectures)
Parameterized Intractability. (9 Lectures)

Textbooks

1. M. CYGAN, F. FOMIN, L. KOWALIK, D. LOKSHTANOV, D. MARX, M. PILIPCZUK, M. PILIPCZUK, S. SAURABH (2015), Parameterized Algorithms, Springer.
2. F. FOMIN, D. LOKSHTANOV, S. SAURABH, M. ZEHAVI (2019), Kernelization: Theory of Parameterized Preprocessing, Cambridge University Press.

Self-Learning Material
1. https://www.youtube.com/watch?v=Ex8TueBsF1g&list=PLhkiT_RYTEU0gpi97fqjtaHy9Gk47oF85&a mp;index=1
2. https://www.youtube.com/watch?v=VAEul4J3Br8&list=PLhkiT_RYTEU1ekkuDwau01N4_Axr_PHQI &index=1

Title	**Principles of Biological Vision and Applications**	Number	**CSL7420**
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech., M.Tech., Ph.D.	Type	Elective
Prerequisite	None	Antirequisite	None

Objectives
The instructor will provide:
An account of various biological perceptions and cognitive processes.
Computational models for such processes
Examples of machine vision applications with such computational models.

Learning Outcomes
On completion of the course, the students are expected to have the ability to:
Appreciate the principles of biological perception and cognition.
Apply computational models for perception and cognition processes of biological vision in various machine vision tasks.

Contents
(***Fractal 1*** **(14 Lectures))**
Introduction: Perception, reasoning and cognition; roles of knowledge, memory and learning. (2 Lectures)
Eye and early vision: Lateral Inhibition; convolution; detection of oriented edges; color, texture and motion perceptions; peripheral vision. (4 Lectures)
Reasoning systems: Deductive, abductive and inductive reasoning; statistical property of nature; Bayesian framework of reasoning; Bayesian networks; parameter estimation; on complexity of models and prior probabilities; information integration; hierarchical Bayesian models and inductive generalization. (8 Lectures)

(***Fractal 2*** **(14 lectures))**
Late Vision: Depth perception; perceptual grouping; foreground-background separation; multi-stability; models for object recognition; hierarchical models; visual quality and aesthetics. (5 Lectures)
Visual attention: cognitive, information-theoretic, Bayesian, context based and object-based attention models; evaluation of attention models. (6 Lectures)
Introduction to cognitive architectures: Active vision; perceive-interpret-act cycles; modeling cognition processes; roles of short-term and long-term memory; review of STAR architecture. (3 Lectures)

(***Fractal 3*** **(14 lectures))**
Knowledge Representation and learning: Role of knowledge in visual interpretation; memory, knowledge and learning; short-term and long-term memory; structured and unstructured knowledge; semantic networks; frame-based representation; symbol grounding problem; perceptual knowledge; unified framework for conceptual and perceptual knowledge; learning in knowledge-based systems. (5 Lectures)
Neural networks for vision: Introduction to neural network; CNN for image classification and object detection; knowledge representation in neural networks; RNN and life-long learning; LSTM; recurrent attention models; reinforcement learning, meta-learning and multi-task learning; knowledge-infused learning; graph networks. (6 Lectures)
Applications in machine vision tasks: Computational photography; digital heritage; social robots; smart content delivery and re-purposing. (3 Lectures)

Textbook
H. GHOSH (In Press – expected Jun/Jul 2020*)*, Computational Models for Cognitive Vision,, Wiley-IEEE Press.

Reference Material
Research literature to be announced in class.

Course Title	**Randomized Algorithms**	Course No.	**CS7xxx**
Department	Computer Science and Engineering	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B.Tech, M. Tech, Ph.D.	Type	Bouquet/Elective
Prerequisite	Design and Analysis of Algorithms, Probability, Statistics and Stochastic Processes	Antirequisite	None

Objectives
This course presents basic concepts in the design and analysis of randomized algorithms.

Learning Outcomes
Students are expected to have the ability to:
1. Understand several of the main thrusts of work in randomized algorithms.
2. Read current research publications in the area.
3. Use randomized algorithm techniques for real world problems.

Contents
Tools and Techniques: Basic probability theory; Randomized complexity classes; Game-theoretic techniques; Markov, Chebyshev, and moment inequalities; Limited independence; Tail inequalities and the Chernoff bound; Conditional expectation; The probabilistic method; Markov chains and random walks; Algebraic techniques; Probability amplification and derandomization. (22 Lectures)
Applications: Sorting and searching; Data structures; Combinatorial optimization and graph algorithms; Geometric algorithms and linear programming; Approximation and counting problems; Parallel and distributed algorithms; Online algorithms. (20 Lectures)

Text Books
R. MOTWANI, P. RAGHAVAN (1995), Randomized Algorithms, Cambridge University Press, 1st Edition.

Reference Books
M. MITZENMACHER, E. UPFAL (2017), Probability and Computing: Randomized Algorithms and Probabilistic Analysis, Cambridge University Press, 2nd Edition.
W. FELLER (2008), An Introduction to Probability Theory and Its Applications, Volumes I and II, John Wiley, 2nd Edition.
P. BILLINGSLEY (2012), Probability and Measure, John Wiley.

Self Learning
https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-856j-randomized-algorithms- fall-2002/lecture-notes/.

Title	**Resource Constrained AI**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech, M.Tech., PhD	Type	Elective
Prerequisite	PRML / Introduction to Machine Learning	Antirequisite	None

Objectives
The Instructor will:
1. Explain the challenges of implementing AI and machine learning algorithms on devices with memory and power constraints
2. Provide methods to reduce computational complexity of AI techniques

Learning Outcomes
The students are expected to have the ability to:
1. Understand the constraints of implementing AI algorithms on limited memory devices
2. Design and develop techniques to reduce inference time memory footprint of machine learning models

Contents
Introduction: Overview and motivation, challenges of resource constrained AI, why AI on edge (4 lectures)
Edge Computing: Edge devices and their limitations, Edge and fog computing, Distributed computing, communication links, communication overhead in IoT devices (8 lectures)
Monitoring: Prediction accuracy, numeric accuracy, precision, memory footprints, computational complexity of AI models (4 lectures)
Memory Optimization of Models: KiloByte-size models, floating-point v/s fixed-point, SeeDot (8 lectures)
Edge AI: Resource-efficient kNN, SVM and deep learning models, Toeplitz matrix, Bonsai, ProtoNN, EMI-RNN, FastRNN, FastGRNN (10 lectures)
Current Trends and Future: Hardware accelerators for Edge AI, Vision Processing Unit (VPU), Streaming Hybrid Architecture Vector Engine (SHAVE), Intel's Movidius Neural Compute Stick (NCS), Open Neural Network Exchange (ONNX), Future trends (10 lectures)

Laboratory Experiments
Implementation of Bonsai, CNN training using SeeDot language etc.

Textbook
1. Alippi, C., (2014), *Intelligence for Embedded Systems: A Methodological Approach*, Springer.

Preparatory Course Material
1. EdgeML by Microsoft, https://github.com/Microsoft/EdgeML/#edge-machine-learning
2. NCSDK by Intel https://github.com/movidius/ncsdk

Course Title	**Security and Its Applications**	Course No.	**CS 7xx**
Department	Computer Science and Engineering	Structure (L-T-P [C])	3-0-0 [3]
Offered for	M.Tech., PhD	Type	Compulsory
Prerequisite	Computer Networks	Anti-requisite	Cyber Security

Objectives

1. Provide the fundamental principles of access control models and techniques, authentication and secure system design
2. Introduce a strong understanding of different cryptographic protocols and techniques and be able to use them
3. Provide methods for authentication, access control, intrusion detection and prevention

Learning Outcomes

1. Understand cryptography and network security concepts and application
2. Identify and investigate network security threat
3. Analyze and design network security protocols

Contents:

Introduction to Cyber Security: *Internet governance – Challenges and constraints, Cyber threats. (2 Lectures)*
Cyber Security Vulnerabilities and Cyber Security Safeguards: *Cyber security, Vulnerabilities, safeguards, Access control, Authentication, Biometrics, Deception, Denial of Service Filters, Ethical hacking, Firewalls, Response, Scanning, Security policy, Threat management. (8 Lectures)*
Cryptography: *Shannon's Approach to Cryptography: Measures of security, Perfect secrecy, Definition of entropy, One-time pad, Symmetric Key Cryptography, Cryptographic Hash Functions, Authentication, Public Key Cryptosystems, Key Distribution and Key Agreement Protocols(4 Lectures)*
Web Security: *Basic security for HTTP applications and services, Basic security for SOAP services, Identity management and Web services, Authorization patterns, Security considerations, challenges. (4 Lectures)*
Network Security: *TCP/IP threats, The IPSEC protocol, The SSL and TLS protocols, Firewalls and Virtual Private Networks (VPNs), Electronic mail security, Worms, DDoS attacks, BGB and security considerations (12 lectures)*
Intrusion Detection and Prevention: *Intrusion detection and Prevention techniques, Anti-malware software, Security information management, Network session analysis, System integrity validation. (4 Lectures)*
Cyber-physical security: *IoT security, sensor actuator network security (4 lectures)*
Block Chain: *Introduction to Blockchain, Blockchain Architecture and Design, Consensus (Byzantine Fault, Proof of Work, Proof of Stake), Permissioned Blockchains, Components of blockchain (5 lectures)*

Text Book

1. Stallings, W. (2017). Cryptography and network security, 7/E. Pearson Education India
2. Douglas R. Stinson, Maura B. Paterson (2018). Cryptography: theory and practice. 4/E Chapman and Hall/CRC

Reference Books

1. Mao, W. (2004). Modern cryptography: theory and practice. Pearson Education India
2. Pfleeger, C. P., & Pfleeger, S. L. (2018). Security in computing. 5/E, Prentice Hall Professional Technical
3. Reference
4. Goldreich, O. (2009). Foundations of cryptography: volume 2, basic applications. Cambridge university press
5. Forouzan, B. A. (2015). Cryptography & network security. 3/E, McGraw-Hill, Inc.

Supplementary Resources:

1. Network and Computer Security - MIT OpenCourseWare https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-857-network-and-computer-security-spring-2014/
2. Cryptography I, Coursera, Stanford University, https://www.coursera.org/learn/crypto

Title	**Social Computing**	Number	CS L7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech, M.Tech, Ph.D	Type	Elective
Prerequisite	Data Structure and Algorithms	Antirequisite	None

Objectives
The Instructor will:
1. Introduce the fundamental and current challenges in social computing.
2. Explain how to design social computing systems to be effective and responsible.

Learning Outcomes
The students will have the ability to:
1. Explore social computing systems, get experience with social data analyses and focus on design, and evaluation of a social software as their final project for the course.
2. Engage in the creation of new computationally-mediated social environments.

Contents
Introduction and overview Social Software: *Introduction,* Different types of social software like blogging, microblogging, Q&A, Forum, Wiki etc. *[3 Lectures]*
Social information processing *:* Tagging, Social Navigation, Social Search, Social Bots *[4 Lectures]*
Group collaboration*:* Wikis and Wikipedia, Computer supported collaboration tools, Content sharing, Open source software development *[6 Lectures]*
Recommender systems *:* Content based, collaborative filtering, challenges of social information processing *[5 Lectures]*
Social data analysis *:* Visualization, Sense-making, APIs, Facebook, Wikipedia, Twitter *[6 Lectures]*
Social capital *:* Definitions and measures, Social capital and social networks, Role of online communities on social capital *[6 Lectures]*
Challenges of online communities*:* Dealing with newcomers, under-contribution problem Encouraging contributions to online communities, Strategies supported by social science theories *[8 Lectures]*
Social computing, cities, and society*: Impact on physical and psychological well-being, Connection to real world, Equality of participation [4 Lectures]*

Text Book
Becker, Howard S. Writing for social scientists: How to start and finish your thesis, book, or article. University of Chicago Press, 2008.

Reference Books
1. Bail et al (2018). Exposure to opposing views on social media can increase political polarization. Proceedings of the National Academy of Sciences, 115(37), 9216-9221.
2. Dubois, E., & Blank, G. (2018). The echo chamber is overstated: the moderating effect of political interest and diverse media. Information, Communication & Society, 21(5), 729-745.
3. Salganik, M. J., & Watts, D. J. (2009). Web-based experiments for the study of collective social dynamics in cultural markets. Topics in Cognitive Science, 1(3), 439-468.

Self-Learning Material
Social Computing, IIT Kharagpur, http://cse.iitkgp.ac.in/~pawang/courses/SC16.html

Title	**Social Network Analysis**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B. Tech.,M.Tech., Ph.D.	Type	Elective
Prerequisite	None	Antirequisite	Social Networks

Objectives
The Instructor will:
1. Introduce the social networks and the research areas therein.
2. Provide with the mathematical foundation required for social network analysis
3. Cover various concepts, terminologies and algorithms related to social network analysis
4. Conduct tutorial sessions to use NetworkX library in Python for network analysis

Learning Outcomes
The students are expected to have the ability to:
1. Understand the applications related to social networks
2. Write program with social network datasets in Python
3. Formulate real-world problems with any relational data set resembling social networks

Contents
Introduction and Different Types of Networks (1 Lecture)
Graph Introduction: Adjacency Matrix, Paths, Connectivity, Incidence Matrix, Distance, Breadth-First-Search, Directed Graph (1 Lecture)
Introduction to Python and NetworkX (1 Lecture)
Network Measures, Centrality, Core, Cliques and Clan, Strong and Weak Ties, Homophily, Structural Balance, Components (4 Lectures)
Network Data Sets and Structural Analysis in Python+NetworkX+Pandas (2 Lectures)
Network Models: Random Networks, Scale Free Networks, The Barabási-Albert Model, Fuzzy-Granular Social Network (4 Lectures)
Generate Synthetic Networks, Using Network Models in Python+NetworkX (2 Lectures)
Game Theory Introduction, Modeling Network Traffic using Games (3 Lectures)
Information Cascades, Small-World Phenomenon, Epidemics (4 Lectures)
Implementing Information Diffusion Algorithms in Python+NetworkX (2 Lectures)
Community Detection (3 Lectures)
Implementing Community Detection Algorithms in Python+NetworkX (2 Lectures)
Link Prediction (2 Lectures)
Implementing Link Prediction Algorithms in Python+NetworkX (2 Lectures)
Evolving Network and Temporal Networks (2 Lectures)
Working with Temporal Network Data (2 Lectures)
Connected Caveman Problem, Link Analysis and Web Search (2 Lectures)
Implementing PageRank algorithm in Python+NetworkX (1)
Network Data Science State-of-the-art (2 Lectures)

Textbook
1. Networks, Crowds, and Markets: Reasoning About a Highly Connected World, by David Easley and Jon Kleinberg, (Cambridge University Press - Sep 2010) – Pre-publication draft available online.
2. Network Science, by Albert-Laszlo Barabasi, (Cambridge University Press - August 2016) -- freely available under the Creative Commons licence.
3. Networks, by Mark Newman, (Oxford University Press, 2nd-edition - Sep 2018)

Reference Books
1. Complex and Adaptive Dynamical Systems, by Claudius Gros, (Springer, 4th Edition - 2015).
2. The Structure of Complex Networks Theory and Applications, by Ernesto Estrada, (Oxford University Press,2011).
3. Exploratory Social Network Analysis with Pajek, by Wouter de Nooy, Andrej Mrvar, and Vladimir Batagelj, (Cambridge University Press, 3rd Edition - July 2018)

Self Learning Material

1. https://www.barabasilab.com/course
2. https://nptel.ac.in/courses/106106169/#

Title	**Soft Computing Techniques**	Course No.	CSL7XXX
Department	Computer Science and Engineering, AI&DS	Structure (L-T-P [C])	3–0–0 [3]
Offered for	B.Tech, M.Tech., Ph.D.	Type	Elective
Prerequisite	None	Antirequisite	None

Objectives
Introduction of different soft computing techniques, their integration and applications.

Learning Outcomes
Students are expected to have the ability to:
1. Identify and describe soft computing techniques
2. Understand soft computing approaches in problem solving
3. Formulate real-world methodologies to data mining using soft computing tools

Contents
Introduction to Soft Computing: Difference between soft and hard computing, Fuzzy Computing, Neural Computing, Genetic Algorithms, Associative Memory, Adaptive Resonance Theory, Applications (4 Lectures)
Fuzzy Sets and Fuzzy Systems: Membership functions, Fuzzy operations, Fuzzy relations, Fuzzy proposition, Fuzzy implication, Fuzzification, Fuzzy inference, Fuzzy rule based systems, Defuzzification (7 Lectures)
Genetic Algorithm: Representation, Fitness function, Population, Operators – Selection, Mutation, Crossover, Others, Multi-objective optimization problems (8 Lectures)
Metaheuristic and Swarm Intelligence: Ant colony optimization, Bee colony optimization, Particle swarm optimization, Cuckoo search algorithm and others (6 Lectures)
Rough Sets, Knowledge representations, Rough decision making and data mining techniques, Granular Computing (5 Lectures)
Hybrid Systems: Neuro-fuzzy systems, Rough-neural computing, Fuzzy logic and Genetic Algorithm, GA based back propagation networks, Fuzzy associative memories, Hybrid systems using fuzzy and rough sets (9 Lectures)
Big Data Challenges and Soft Computing Opportunity: Uncertainties in Big Data Inputs, Uncertainties in Big Data Decisions (3 Lectures)

Reference Books

1. S.N. SIVANANDAM, S.N. DEEPA (2018), Principles of Soft Computing, Wiley India, 2018, 3rd Edition.
2. F.O. KARRAY, C. De SILVA (2004), Soft Computing and Intelligent Systems Design: Theory, Tools and Applications, Pearson Education.
3. Z. PAWLAK (1991), Rough Sets: Theoretical Aspects of Reasoning about Data, Springer Netherlands.

Self Learning Material
1. Computer Science and Engineering - NOC: Introduction to Soft Computing: https://nptel.ac.in/courses/106105173/
2. Soft Computing — IT60108: http://cse.iitkgp.ac.in/~dsamanta/courses/sca/index.html.
3. Soft Computing: http://www.myreaders.info/html/soft_computing.html.

Course Title	**Software Defined Networks**	Course No.	**CS7XXX**
Department	Computer Science and Engineering, Electrical Engineering	Structure (L-T-P-C)	2-0-0
Offered for	B.Tech	Type	Elective
Pre-requisites	Computer Networks	Antirequisite	None

Objectives
The course is intended to provide students with an understanding of the basics of SDNs.

Learning Outcomes
The course would enable students to look forward to SDN applications such as in datacenters and 5G systems.

Contents
Networking basics: Switching, Addressing, Routing (2 lecture)
Switching Architecture: Data, control and management planes, hardware lookup, forwarding rules, dynamic forwarding tables, autonomous switches and routers (4 lectures)
SDN Architectures: Plane Separation, Simple Device and Centralized Control, Network Automation and Virtualization, Openness SDN Controllers, SDN Applications, Northbound and Southbound APIs (5 lectures) **OpenFlow:** Switch-Controller Interaction, Flow Table, Packet Matching, Actions and Packet Forwarding, Extensions and Limitations (3 lectures)
Network Function Virtualization: SDN vs. NFV, OPNFV, Inline Network Functions, Service Creation and Chaining, NFV Orchestration (5 lectures)
Emerging SDN Models: Protocol Models, Controller Models, Application Models, SDN in Datacenters: Multitenancy, Failure Recovery, SDN in Internet eXchange Points (IXPs) (4 lectures)
Data Center Networking in the context of SDN (5 lectures)

Note
Please note that the course material would not be based on a single book or resource. It would be a combination of textbook material, research papers and other sources from the Internet.

Textbooks
P. GORANSSON, C. BLACK, T. CULVER (2016), Software Defined Networks: A Comprehensive Approach. Morgan Kaufmann.
K. GRAY, T.D. NADEAU (2016), Network Function Virtualization, Morgan Kaufmann.

Self-learning Material
Relevant research papers.

Course Title	**Software Testing and Quality Assurance**	Course No.	**CS7xxx**
Department	Computer Science and Engineering	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B.Tech, M.Tech, Ph.D.	Type	Elective
Prerequisite	Software Engineering	Antirequisite	None

Objectives
The course provides an introduction to the software engineering testing process, describes the quality assurance process and its role in software engineering.

Learning Outcomes
The student will be able to work with various testing techniques, methods and tools.
The student will be able to demonstrate proficiency in managing an industry scale software project to customer requirements.
Students will be able to describe the state of the art validation and verification techniques.

Contents
Overview and Basics: Software quality, Quality assurance and its context, Quality Engineering. (6 Lectures)
Software Testing: Concepts, Issues, and Techniques, Testing process, Testing levels such as unit, module, subsystem, system; Automatic and Manual techniques for generating and validating test data, Static vs dynamic analysis, Functional testing, Web application testing, Reliability assessment. (20 Lectures)
Quality Assurance beyond testing: Defect prevention and Process improvement, Software inspection, Formal verification, Fault tolerance and Failure containment, Comparing quality assurance techniques and activities. (8 Lectures)
Quantifiable Quality Improvement: Feedback loop and activities for Quantifiable quality improvement, Quality model and measurements, Defect classification and analysis, Risk identification, Reliability engineering. (8 Lectures)

Reference Books
J. TIAN (2005), Software Quality Engineering: Testing, Quality Assurance and Quantifiable Improvement, Wiley.
P. AMMANN, J. OFFUTT (2016), Introduction to Software Testing, Cambridge University Press, 2nd Edition.

Course Title	**Speech Understanding**	Course No.	**CSL7xx**
Department	CSE / EE	Structure (L-T-P-C)	3-0-0 [3]
Offered for	B.Tech CSE, AI&DS	Type	Elective
Prerequisite	PRML/IML/ML	Antirequisite	None

Objectives

1. To provide insights into fundamental concepts and algorithms related to speech processing and understanding
2. Impart working expertise by introducing practical problems.

Learning Outcomes

Students will have the ability to:

1. Build a speech recognition system
2. Design and implement basic speech based application

Contents

Introduction to Speech processing: Digitization and Recording of speech signal, Review of Digital Signal Processing Concepts, Human Speech production, Acoustic Phonetics and Articulatory Phonetics, Different categories speech sounds and Location of sounds in the acoustic waveform and spectrograms. (14 Lectures)

Speech recognition: Analysis and Synthesis of Pole-Zero Speech Models, Short-Time Fourier Transform, Analysis:- FT view and Filtering view, Synthesis:-Filter bank summation (FBS) Method and OLA Method, Features Extraction, Extraction of Fundamental frequency, Speech Enhancement, Clustering and Gaussian Mixture models, Speaker Recognition.(14 Lectures)

Speech based applications: HMM and Neural models for speech recognition, Speech generation, Question answering, Dialogue systems, Other Speech based Applications.(14 Lectures)

Text Books

1. T.F. QUATIERI (2002), Discrete-Time Speech Signal Processing, Prentice-Hall, New Jersey.
2. D. JURAFSKY, J.H. MARTIN, Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition (3rd Edition Draft), 2019.

Reference Books

1. Y. GOLDBERG (2016), A Primer on Neural Network Models for Natural Language Processing, Journal of Artificial Intelligence Research.
2. I. GOODFELLOW, Y. BENGIO, A. COURVILLE (2016), Deep Learning, The MIT Press, 1st Edition.
3. S.K. PATRA (2011), Digital Signal Processing: A Computer-Based Approach, McGraw-Hill, 4th Edition.

Title	**Sustainable Computing**	Number	CS L7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech (All), M.Tech	Type	Elective
Prerequisite	PRML / Introduction to Machine Learning	Antirequisite	None

Objectives
The Instructor will:
Describe ways to leverage networking, sensing, and computational strategies to help address sustainability issues related to biodiversity, climate, environment, urban design, transportation, buildings and others.
Draw from professional experiences to ground all topics and discussions in "real world" examples.

Learning Outcomes
The students will have the ability to:
Gain an understanding of how data and computation can be used to identify root causes of sustainability challenges, create targeted strategies to address them, engage stakeholders, and track and communicate progress.
Develop skills to design metrics and solutions for evaluation of progress towards sustainability goals.
Comprehend the power and shortcomings of data in communicating complex sustainability issues.

Contents
Introduction, SDG goals, & Sustainability metrics: Humanity and the environment, Sustainability, Dimensions of sustainable design, Sustainability metrics. [3 Lectures]
Climate: Computational techniques for weather predictions & models of climate variance. [5 Lectures]
Biosphere: Species distribution modelling; Aggregate phenomena modelling; Ecosystem modelling; Animal identification; Case study: Green Security games, Bird Returns program. [6 Lectures]

Health, Food & Water: Health/medical informatics & Telemedicine; Poverty mapping, Food & Farm optimization; Modelling for water distribution. [6 Lectures]
Insuring impact with technology for Education & Gender equality. [3 Lectures]
Usability & HCI for assistive technologies. [3 Lectures]
Green technology & E-waste. [2 Lectures]

Sustainable energy systems: Energy-constrained scheduling, Electricity demand & renewable resource prediction, Models for energy consumption, Smart grid. [4 Lectures]
IT energy efficiency: Measure energy efficiency of computer system components in sync with solution frameworks; Energy consumption estimation of whole systems; Energy efficiency metrics; Sustainable resource management techniques for cloud computing, edge/fog computing, high-performance computing, wearable computing, Internet-of-Things (IoT), and cyber-physical systems; Sustainable data centers. [10 Lectures]

Text Books
1. Theis, T., & Tomkin, J. (2015). Sustainability: A Comprehensive Foundation. OpenStax CNX.
2. Issa, T., & Isaias, P. (2015). Sustainable Design: HCI, Usability and Environmental Concerns. Springer-Verlag London.
3. Green & Sustainable Computing - Part I & II. Elsevier.

Reference Material
1. Gomes, C. et al. (2019). Computational Sustainability: Computing for a Better World and a Sustainable Future. Communications of the ACM, 62(9),56-65.
2. Sustainability Metrics Reading List: https://afddf8e8-2dfc-4526-9ba6-71e1899413f3.filesusr.com/ugd/571f98_b203aa22d378452b986dbc667acb13b7.pdf.
3. Reading List: http://www.cs.cornell.edu/courses/cs6702/2011sp/.
4. Reading List: https://www.coursera.org/learn/sustainability#syllabus

Title	**Stream Analytics**	Number	**CS7xxx**
Department	Computer Science	L-T-P [C]	3–0–0 [3]
Offered for	B. Tech., M.Tech., PhD	Type	Elective
Prerequisite	Introduction to Machine Learning, PRML	Antirequisite	None

Objectives
Provide background on some of the important models, algorithms, and applications related to stream data.

Learning Outcomes
Ability to understand and apply the practical and algorithmic aspects related to various topics of data streams

Contents
Introduction: Stream and mining algorithms. (2 Lectures)
Clustering Massive Data Streams: Micro-clustering based stream mining, Clustering evolving data streams, Online Micro-cluster maintenance, High-dimensional projected stream clustering, Classification of data streams using micro-clustering, On-demand stream classification, Applications of micro- clustering. (12 Lectures)
Classification Methods in Data Streams: Ensemble based classification, Very fast decision trees, On-demand classification, Online Information Network. (6 Lectures)
Distributed Mining of Data Streams: Outlier and anomaly detection, Clustering, Frequent itemset mining, Classification, Summarization. (6 Lectures)
Change Diagnosis Algorithms in Evolving Data Streams: Velocity density method, Clustering for characterizing stream evolution. (4 Lectures)
Multidimensional Analysis of Data Streams using Stream Cubes: Architecture for online analysis of data streams, Stream data cube computation, Performance study. (6 Lectures)
Dimensionality Reduction and Forecasting on Streams: Principal Component Analysis, Auto-regressive models and recursive least squares, Tracking correlations and hidden variables. (6 Lectures)

Text Book
C.C. AGGARWAL, (Ed.), Data Stream: Models and Algorithms, Kluwer Academic Publishers, 2007.

Title	**Vehicular Ad-Hoc Networks (VANETs)**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B. Tech, M.Tech. and Ph.D.	Type	Elective
Prerequisite	Computer Networks, Introduction to Wireless Ad hoc Networks	Antirequisite	None

Objectives
The Instructor will:
1. Introduce the emerging technologies, standards and applications in vehicular communication systems.
2. Provide the design considerations and challenges of vehicle-to-infrastructure and vehicle-to-vehicle communications. Theories such as vehicular mobility modeling, and vehicular technologies and standards from the physical to network layers will be introduced in the course. Examples of emerging applications of vehicular communications in Intelligent Transportation Systems will also be studied and discussed.

Learning Outcomes
The students are expected to have the ability to:
1. Understand and describe the basic theories and principles, technologies, standards, and system architecture of vehicular ad-hoc networks (VANET) or inter-vehicle communication networks.
2. Analyze, design, and evaluate vehicular communication platforms for various kinds of safety and infotainment applications.

Contents
Introduction: Basic principles and challenges, past and ongoing VANET activities (2 Lectures)
Cooperative Vehicular Safety Applications: Enabling technologies, cooperative system architecture, safety applications (2 lectures)
Vehicular Mobility Modeling: Random models, flow and traffic models, behavioral models, trace and survey based models, joint transport and communication simulations (4 lectures)
Physical Layer Considerations for Vehicular Communications: Signal propagation, Doppler spread and its impact on OFDM systems (4 lectures)
MAC Layer of Vehicular Communication Networks: Proposed MAC approaches and standards, IEEE 802.11p (8 lectures)
VANET Routing protocols: Opportunistic packet forwarding, topology-based routing, geographic routing (8 lectures)
Emerging VANET Applications: Limitations, example applications, communication paradigms, message coding and composition, data aggregation (8 lectures)
Standards and Regulations: Regulations and Standards, DSRC Protocol Stack, Cellular V2X (6 lectures)

Laboratory Experiments
Programming exercises using NS3, QualNet and Java

Textbook

1. Olariu, S., & Weigle, M. C. (2017). *Vehicular networks: from theory to practice*. Chapman and Hall/CRC
2. Murthy, C. S. R. (2006). *Ad hoc wireless networks: Architectures and protocols*. Pearson Education India

Reference Books

1. Emmelmann, M., Bochow, B., & Kellum, C. (Eds.). (2010). *Vehicular networking: Automotive applications and beyond* (Vol. 2). John Wiley & Sons
2. Claudia Campolo , Antonella Molinaro, Riccardo Scopigno(2015). Vehicular ad hoc Networks, Springer
3. Hartenstein, H., & Laberteaux, K. (2010). *VANET: vehicular applications and inter-networking technologies* (Vol. 1). Chichester: Wiley
4. Sommer, C., & Dressler, F. (2015). *Vehicular networking*. Cambridge University Press

5. Moustafa, H., & Zhang, Y. (2009). *Vehicular networks: techniques, standards, and applications*. Auerbach publications

Self Learning Material

1. Center for Autonomous Intelligent Networks and Systems (CAINS), University of California, Los Angeles (UCLA),http://www.cains.cs.ucla.edu/

Title	**Video Processing**	Course No.	CSL7XX
Department	CSE, EE	Structure (L-T-P [C])	3–0–0 [3]
Offered for	B.Tech., M.Tech., Ph.D.	Type	Elective
Prerequisite	Linear Algebra	Antirequisite	None

Objectives
1. To make the students familiar with several issues and challenges involved in the task of video processing
2. To enable students to formulate problems related to video processing and explore solutions

Learning Outcomes
The students are expected to have the ability to:
1. Address the challenging issues of video processing and to come with new solutions by their own
2. Handle multi-dimensional signals other than videos

Contents
Digital Images and Video:Human Visual Systems, Analog Video, Digital Video, 3D Video, Video Quality (2 Lectures)
Multi Dimensional Signals and Systems: Multi-dimensional Signals, Multi-dimensional Systems, Multi-
dimensional transforms, Multi-dimensional Sampling Theory (7 Lectures)
Motion Estimation: Camera Models, Motion Models, Motion Estimation, Differential Methods, Matching Methods, Non-linear Optimization Methods, 3-D Motion and Shape Estimation (10 Lectures)
Video Segmentation and Tracking: Basics of Segmentation, Video based Segmentation Algorithms Change Detection, Motion Segmentation, Motion Tracking, Performance Evaluation (10 Lectures)
Video Filtering:Spatio-temporal Filtering, Video Format Conversion, Multi-Frame Noise Filtering, Multi- Frame Restoration (6 Lectures)
Video Compression: Motion JPEG 2000, MPEG-4, HEVC, SHVC, H.264 (3 Lectures)
Modern Topics in Video Processing: Ego-centric Video Processing, 360-degree Video, Streaming Video (4 Lectures)

Textbooks
1. A.M. TEKALP (2015), Digital Video Processing, Prentice Hall Signal Processing Series, 2nd Edition.
2. A. BOVIK (2009), The Essential Guide to Video Processing, Academic Press, 2nd Edition.

Reference Books
1. E. MAGGIO, A. CAVALLARO (2011), Video Tracking: Theory and Practice, Wiley and Sons.
2. M. WOHL (2017), The 360° Video Handbook.
3. Research literature

Course Title	**Virtualization and Cloud Computing**	Course No.	**CS7xx**
Department	Computer Science & Engineering	Structure (L-T-P [C])	3-0-0 [3]
Offered for	B.Tech	Type	Elective
Prerequisite	Operating Systems, Computer Networks	Antirequisite	None

Objectives
This course will introduce fundamentals of virtualization and concepts, frameworks, and applications in Cloud Computing to the audience

Learning Outcomes
1. To understand various virtualization techniques and their features and limitations
2. To understand design and concepts of a cloud computing framework

Contents
Cloud Computing: Concept, Definition, Cloud Types and Service Deployment Models. (5 Lectures)
Virtualization: Concept, Definition, Types of Virtualization, Hardware Virtualization, Full and Para Virtualization, Hypervisors, Hardware-assisted virtualization, operating system level virtualization, application virtualization (10 Lectures)
Virtual and Physical Networking: Introduction, vSwitches, virtual NICs, Virtual Networking, virtual LAN (5 Lectures)
Storage Virtualization: Introduction, SAN/NAS versus storage virtualization. (4 Lectures)
Virtual Machine Management: Base Virtual Machine, Virtual CPUs, Sockets, Cores, Memory Scaling Up and Scaling Down, USB Support, Virtual Disks, Live Migration. Security. (10 Lectures)
Containers: Concept, Definition, Docker, Container versus Virtualization, Portability, Remote deployment (5 Lectures)
Applications and Case Studies: Linux KVM, VirtualBox, Openstack. (3 Lectures)

Text Book
1. D.E. SARNA (2010), Implementing and Developing Cloud Computing Applications, CRC Press.
2. B.S. SODHI (2017), Topics in Virtualization and Cloud Computing, Ropar PB India, 2017.

Reference Book
B. FURHT, A. ESCALANTE (2010), Handbook of Cloud Computing (Vol. 3), Springer.

700 - Level Courses

- ## Selected / Special Topics (CSE and AI&DS)

Title	**Selected Topics in Artificial Intelligence - I**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	1–0–0 [1]
Offered for	B.Tech., M.Tech., PhD	Type	Elective
Prerequisite	Decided by the instructor		

Objectives

The Instructor will:

1. Expose the students to the latest upcoming fields in the area of artificial intelligence

Learning Outcomes

The students are expected to have the ability to:

1. Apply the knowledge of recent topics to specific research areas in the field of artificial intelligence

Contents

The topic clouds for the course include contemporary topics in artificial intelligence and may be updated according to the instructor.

Textbook

Relevant Textbook and/or research papers to be announced by the instructor.

Self-Learning Material

Relevant Textbook and/or research papers to be announced by the instructor.

Preparatory Course Material

Relevant Textbook and/or research papers to be announced by the instructor.

Title	**Selected Topics in Artificial Intelligence - II**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	2–0–0 [1]
Offered for	B.Tech., M.Tech., PhD	Type	Elective
Prerequisite	Decided by the instructor		

Objectives
The Instructor will:
1. Expose the students to the latest upcoming fields in the area of artificial intelligence

Learning Outcomes
The students are expected to have the ability to:
1. Apply the knowledge of recent topics to specific research areas in the field of artificial intelligence

Contents
The topic clouds for the course include contemporary topics in artificial intelligence and may be updated according to the instructor.

Textbook
Relevant Textbook and/or research papers to be announced by the instructor.

Self-Learning Material
Relevant Textbook and/or research papers to be announced by the instructor.

Preparatory Course Material
Relevant Textbook and/or research papers to be announced by the instructor.

Title	**Selected Topics in Artificial Intelligence - III**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [1]
Offered for	B.Tech., M.Tech., PhD	Type	Elective
Prerequisite	Decided by the instructor		

Objectives
The Instructor will:
1. Expose the students to the latest upcoming fields in the area of artificial intelligence

Learning Outcomes
The students are expected to have the ability to:
1. Apply the knowledge of recent topics to specific research areas in the field of artificial intelligence

Contents
The topic clouds for the course include contemporary topics in artificial intelligence and may be updated according to the instructor.

Textbook
Relevant Textbook and/or research papers to be announced by the instructor.

Self-Learning Material
Relevant Textbook and/or research papers to be announced by the instructor.

Preparatory Course Material
Relevant Textbook and/or research papers to be announced by the instructor.

Title	**Selected Topics in Computer Science - I**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	1–0–0 [1]
Offered for	B.Tech., M.Tech., PhD	Type	Elective
Prerequisite	Decided by the instructor		

Objectives
The Instructor will:
1. Expose the students to the latest upcoming fields in the area of computer science

Learning Outcomes
The students are expected to have the ability to:
1. Apply the knowledge of recent topics to specific research areas in the field of computer science

Contents
The topic clouds for the course include contemporary topics in computer science and may be updated according to the instructor.

Textbook
Relevant Textbook and/or research papers to be announced by the instructor.

Self-Learning Material
Relevant Textbook and/or research papers to be announced by the instructor.

Preparatory Course Material
Relevant Textbook and/or research papers to be announced by the instructor.

Title	**Selected Topics in Computer Science - II**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	2–0–0 [2]
Offered for	B.Tech., M.Tech., PhD	Type	Elective
Prerequisite	Decided by the instructor		

Objectives
The Instructor will:
1. Expose the students to the latest upcoming fields in the area of computer science

Learning Outcomes
The students are expected to have the ability to:
1. Apply the knowledge of recent topics to specific research areas in the field of computer science

Contents
The topic clouds for the course include contemporary topics in computer science and may be updated according to the instructor.

Textbook
Relevant Textbook and/or research papers to be announced by the instructor.

Self-Learning Material
Relevant Textbook and/or research papers to be announced by the instructor.

Preparatory Course Material
Relevant Textbook and/or research papers to be announced by the instructor.

Title	**Selected Topics in Computer Science - III**	Number	CSL7XX0
Department	Computer Science and Engineering	L-T-P [C]	3–0–0 [3]
Offered for	B.Tech., M.Tech., PhD	Type	Elective
Prerequisite	Decided by the instructor		

Objectives

The Instructor will:

1. Expose the students to the latest upcoming fields in the area of computer science

Learning Outcomes

The students are expected to have the ability to:

1. Apply the knowledge of recent topics to specific research areas in the field of computer science

Contents

The topic clouds for the course include contemporary topics in computer science and may be updated according to the instructor.

Textbook

Relevant Textbook and/or research papers to be announced by the instructor.

Self-Learning Material

Relevant Textbook and/or research papers to be announced by the instructor.

Preparatory Course Material

Relevant Textbook and/or research papers to be announced by the instructor.

Title	**Special Topics in Algorithms**	Number	**CS7xxx**
Department	Computer Science	L-T-P [C]	3–0–0 [3]
Offered for	B. Tech., M.Tech., PhD	Type	Elective
Prerequisite	Algorithm Design and Analysis, Maths for Computing	Antirequisite / Preferred Knowledge	None

Objectives

1. The objective of the course is to introduce several advanced algorithmic techniques.

Learning Outcomes

Students will gain the ability to:

1. Learn a new set of techniques to cope with NP-hard problems.
2. Identify novel and significant open research questions in the field.

Contents

Parameterized Algorithms [13 lectures]: Introduction to Parameterized Complexity and basics [2 Lectures]; Branching [4 Lectures]; Iterative Compression [3 Lectures]; Kernelization [4 Lectures]
Approximation Algorithms: [10 lectures]: Greedy Algorithm – Load Balancing, Center Selection Problem, Set Cover [5 Lectures]; The Pricing Method: Vertex Cover, Linear Programming and Rounding: An application to Vertex Cover, Knapsack [5 Lectures]
Randomized Algorithms [10 lectures]: Contention Resolution, Global Mincut, Random Variables and Expectations, Max-3-SAT approximation [7 Lectures]; Color Coding [3 Lectures]
Exact Exponential Time Algorithms [7 lectures]: Exact Algorithms for Coloring, SAT, Directed Feedback Arc Set, Max-Cut, Monotone-Local-Search, Or some other topics of contemporary interest.
Streaming Algorithms [2 lectures]: Introduction to streaming algorithms and its application to some graph theoretic problems.

Textbooks

1. Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Daniel Marx, Marcin Pilipczuk, Michal Pilipczuk, Saket Saurabh (2015): Parameterized Algorithms, Springer.
2. Jon Kleinberg, Eva Tardos (2005), Algorithm Design, Pearson Education, 1st Edition.
3. Fedor V. Fomin, Dieter Kratsch (2010), Exact Exponential Time Algorithms, An EATCS Series, Springer.

Self Learning Material

1. https://www.youtube.com/watch?v=Ex8TueBsF1g&list=PLhkiT_RYTEU0gpi97fqjtaHy9Gk47oF85&index=1
2. https://sites.google.com/view/sakethome/teaching/parameterized-complexity?authuser=0
3. https://www.youtube.com/watch?v=S8Acu3EpvsE&list=PLhkiT_RYTEU2itsMqCNdXUg4cdFUWJn3-&index=4
4. https://www.youtube.com/watch?v=jNfQ3GZlrjM&list=PLhkiT_RYTEU3vSaVleEm_-bIPBzCqRQHK