BIRLA INSTITUTE OF TECHNOLOGY
MESRA
RANCHI, INDIA

CHOICE BASED CURRICULUM

Computer Science and Engineering

P.G Programme

(M. Tech in Computer Science and Engineering)

Department of Computer Science & Engineering

Birla Institute of Technology, Mesra, Ranchi - 835215 (India)

Institute Vision

To become a Globally Recognized Academic Institution in consonance with the social, economic and ecological environment, striving continuously for excellence in education, research and technological service to the National needs.

Institute Mission

- To educate students at Undergraduate, Postgraduate Doctoral and Post-Doctoral levels to perform challenging engineering and managerial jobs in industry.
- To provide excellent research and development facilities to take up Ph.D. programmes and research projects.
- To develop effective teaching and learning skills and state of art research potential of the faculty.
- To build national capabilities in technology, education and research in emerging areas.
- To provide excellent technological services to satisfy the requirements of the industry and overall academic needs of society.

Department Vision

The department strives to be recognized for outstanding education and research, leading to excellent professionals and innovators in the field of Computer Science and Engineering, who can positively contribute to the society.

Department Mission

- To impart quality education and equip the students with strong foundation that could make them capable of handling challenges of the new century.
- To maintain state of the art research facilities and facilitate interaction with world's leading universities, industries and research organization for constant improvement in the quality of education and research.

Program Educational Objectives (PEOs) - Computer Science

PEO 1: Students are trained in such a way that makes them capable of exploiting and enhancing theoretical and practical knowledge in various domains of Computer Science.

PEO 2: Students are imparted with strong base of knowledge that makes them suitable for both industry teaching and research.

PEO 3: Students are trained to develop practical and efficient solutions to the challenges in the growing field of software industry to gain leadership positions in their organization and/or teams.

PEO 4: Students are inculcated with the sensitivity towards ethics, public policies and their responsibilities towards the society to gain trust and respect of others as ethical team members.

PROGRAM OUTCOMES (POs)
M. Tech. in Computer Science and Engineering

PO1: An ability to independently carry out research /investigation and development work to solve practical problems.

PO2: An ability to write and present a substantial technical report/document.

PO3: Students should be able to demonstrate a degree of mastery over the area as per the specialization of the program. The mastery should be at a level higher than the requirements in the appropriate bachelor program.

PO4: In depth understanding of fundamental principles and concepts of various domains of Computer Science.

PO5: Ability for analytical and critical thinking in order to analyse, design and improve existing tools and techniques.

PO6: Knowledge of contemporary issues in the field of Computer Science and ability to engage in lifelong learning.

COURSE INFORMATION SHEET

Course code: CS501
Course title: MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE
Pre-requisite(s): Discrete Mathematics
Co- requisite(s):
Credits: L: 3 T: 0 P: 0
Class schedule per week: 03
Class: M. Tech
Semester / Level: I/5
Branch: Computer Science and Engineering
Name of Teacher:

Course Objectives

This course enables the students to:

1.	Present basic concepts and techniques of linear algebra, probability, statistics and graph theory
2.	Develop mathematical thinking and problem-solving skill
3.	Provide the foundations of probabilistic and statistical analysis
4.	Explain graphs to formulate computational problems

Course Outcomes

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

CO1	Demonstrate skills in solving mathematical problems
CO2	Apply knowledge of computing and mathematics appropriate to the discipline
CO3	Analyze problems and identify the computing requirements appropriate to its solution
CO4	Explain basic concepts in probability theory and statistical analysis
CO5	Articulate the advanced courses in Computer science such as Coding Theory, Artificial Intelligence, Numerical Computation, etc.

SYLLABUS

Module I:
Linear Algebra
Introduction: Matrices and solving set of linear equations, Vector space, Subspace, Linear combination of vectors, Linear dependence and independence of vectors, Bases and dimensions.

(8L)

Module II:
Inner product spaces, Orthogonal vectors and dual vectors, Eigen values and Eigen vectors, Linear programming.

(8L)

Module III:
Probability and Statistics
Frequency distribution and measures of central tendency mean, median mode, quartiles, measures of dispersions and skewness, standard deviation, mean deviation, coefficient of variation, moments.

(8L)

Module IV:
Probability: definition, Distribution: discrete and continuous, Chi-square test, t-test.

(8L)

Module V:
Graph Theory
Introduction: Graphs and its types, Representation of graphs: Adjacency matrix, Incidence matrix, Adjacency list, Planar graph, Kuratowski's Graphs, Clique and maximum Clique finding algorithms.

(8L)

Books recommended:

TEXT BOOK

1. K. Haffman, and R. Kunze, "Linear Algebra", 2nd Edition, Pearson, 2015. **(T1)**
2. G. Williams, "Linear Algebra with Applications", 4th Edition, John & Bartlett. **(T2)**
3. W. Navidi, "Statistics for Engineers and Scientists", 2nd Edition, TMH, 2008. **(T3)**
4. J.K. Goyal, and J. N. Sharma, "Mathematical Statistics", Krishna Prakashan, 2017. **(T4)**
5. Narasingh Deo, "Graph Theory with Applications to engineering and Computer Science", Prentice Hall of India, 2001. **(T5)**
6. Douglas B. West, "Introduction to Graph theory", Pearson Education, 2002. **(T6)**

Course Evaluation:
Individual assignment, Theory (Quiz and End semester) examinations

Gaps in the syllabus (to meet Industry/Profession requirements):

POs met through Gaps in the Syllabus:

Topics beyond syllabus/Advanced topics/Design:

POs met through Topics beyond syllabus/Advanced topics/Design: Teaching through research papers.

Course Delivery Methods

CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Assignments/Seminars
CD3	Laboratory experiments/teaching aids
CD4	Industrial/guest lectures
CD5	Industrial visits/in-plant training
CD6	Self- learning such as use of NPTEL materials and internets
CD7	Simulation

MAPPING BETWEEN COURSE OUTCOMES AND PROGRAM OUTCOMES

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	2	1	1	1	1
CO2	3	3	1	1	1	1
CO3	2	3	2	1	1	1
CO4	3	2	1	1	1	1
CO5	3	1	1	1	1	1

If satisfying and < 34% = L, 34-66% = M, > 66% = H

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,CD6,CD7
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD3,CD4,CD5,CD7

COURSE INFORMATION SHEET

Course code: CS502
Course title: ADVANCED DATA STRUCTURES
Pre-requisite(s): Data Structures, Algorithm Analysis
Co- requisite(s):
Credits: L: 3 T: 0 P: 0
Class schedule per week: 03
Class: M. Tech
Semester / Level: I/5
Branch: Computer Science and Engineering
Name of Teacher:

Course Objectives

This course enables the students:

1.	The student should be able to choose appropriate data structures, understand the ADT/libraries, and use it to design algorithms for a specific problem.
2.	Students should be able to understand the necessary mathematical abstraction to solve problems.
3.	To familiarize students with advanced paradigms and data structure used to solve algorithmic problems.
4.	Student should be able to come up with analysis of efficiency and proofs of correctness.

Course Outcomes

After the completion of this course, students are expected to:

CO1	Understand the implementation of symbol table using hashing techniques.
CO2	Develop and analyze algorithms for red-black trees, B-trees and Splay trees.
CO3	Develop algorithms for text processing applications.
CO4	Interpret the basic working of advanced heaps.
CO5	Appraise the implementation of symbol table using hashing techniques.

SYLLABUS

Module I:
Dictionaries: Definition, Dictionary Abstract Data Type, Implementation of Dictionaries. Hashing: Review of Hashing, Hash Function, Collision Resolution Techniques in Hashing, Separate Chaining, Open Addressing, Linear Probing, Quadratic Probing, Double Hashing, Rehashing, Extendible Hashing.

(8L)

Module II:
Skip Lists: Need for Randomizing Data Structures and Algorithms, Search and Update Operations on Skip Lists, Probabilistic Analysis of Skip Lists, Deterministic Skip Lists

(8L)

Module III:
Trees: Binary Search Trees, AVL Trees, Red Black Trees, 2-3 Trees, B-Trees, Splay Trees

(8L)

Module IV:
Heaps: Balanced Search Trees as Heaps, Array-Based Heaps, Heap-Ordered Trees and Half-Ordered Trees, Leftist Heaps, Skew Heaps, Binomial Heaps, Changing Keys in Heaps, Fibonacci Heaps, Heaps of Optimal Complexity, Double-Ended Heap Structures and Multidimensional Heaps, Heap-Related Structures with Constant-Time Updates.

(8L)

Module V:
Text Processing: Sting Operations, Brute-Force Pattern Matching, The Boyer-Moore Algorithm, The Knuth-Morris-Pratt Algorithm, Standard Tries, Compressed Tries, Suffix Tries, The Huffman Coding Algorithm, The Longest Common Subsequence Problem (LCS), Applying Dynamic Programming to the LCS Problem.

(8L)

Books recommended:

TEXT BOOK

1. Mark Allen Weiss, "Data Structures and Algorithm Analysis in C++," 2nd Edition, Pearson, 2004. **(T1)**
2. Peter Brass, "Advanced Data Structures," Cambridge University Press, 1st Edition. **(T2)**
3. M T Goodrich, & Roberto Tamassia, "Algorithm Design," John Wiley, 2002. **(T3)**

Course Evaluation:
Individual assignment, Theory (Quiz and End semester) examinations

Gaps in the syllabus (to meet Industry/Profession requirements):

POs met through Gaps in the Syllabus:

Topics beyond syllabus/Advanced topics/Design:

POs met through Topics beyond syllabus/Advanced topics/Design: Teaching through research papers.

Course Delivery Methods

CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Assignments/Seminars
CD3	Laboratory experiments/teaching aids
CD4	Industrial/guest lectures
CD5	Industrial visits/in-plant training
CD6	Self- learning such as use of NPTEL materials and internets
CD7	Simulation

MAPPING BETWEEN COURSE OUTCOMES AND PROGRAM OUTCOMES

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	1	1	1	1	1
CO2	2	3	3	1	1	1
CO3	2	3	3	2	1	1
CO4	3	1	1	1	1	1
CO5	3	1	1	1	1	1

If satisfying and < 34% = L, 34-66% = M, > 66% = H

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,CD6,CD7
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD3,CD4,CD5,CD7

COURSE INFORMATION SHEET

Course code: CS504
Course title: DISTRIBUTED SYSTEMS
Pre-requisite(s): Data Structure, Operating system
Co- requisite(s):
Credits: L: 3 T: 0 P: 0
Class schedule per week: 03
Class: M. Tech
Semester / Level: I/5
Branch: Computer Science and Engineering
Name of Teacher:

Course Objectives

This course enables the students to:

1.	Identifying trends in distributed systems
2.	Introducing peer to peer services and distributed file systems
3.	Understanding the issues in process and resource management
4.	Understanding the requirements for designing and supporting distributed systems

Course Outcomes

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

CO1	Understand trends in distributed systems
CO2	Knowledge of peer to peer services and distributed file systems
CO3	Solve issues in process and resource management
CO4	understand the requirements for designing and supporting distributed systems
CO5	Ability to understand the working of distributed systems

SYLLABUS

Module I:
Introduction: Introduction - Examples of Distributed Systems-Trends in Distributed Systems - Focus on resource sharing - Challenges. Case study: World Wide Web.

(8L)

Module II:
Communication in Distributed system: System Model – Inter Process Communication – the API for internet protocols – External data representation and Multicast communication.
Network virtualization: Overlay networks.
Case study: MPI
Remote Method Invocation and Objects: Remote Invocation - Introduction - Request-reply protocols - Remote procedure call - Remote method invocation.
Case study: Java RMI – Group communication - Publish-subscribe systems - Message queues - Shared memory approaches -Distributed objects - Case study: Enterprise Java Beans -from objects to components

(8L)

Module III:
Peer-to-Peer Services: Peer-to-peer Systems - Introduction - Napster and its legacy - Peer-to-peer - Middleware - Routing overlays.
Overlay case studies: Pastry, Tapestry- Distributed File Systems –Introduction File service architecture - Andrew File system.
File System: Features-File model -File accessing models- File sharing semantics
Naming: Identifiers, Addresses, Name Resolution - Name Space Implementation - Name Caches - LDAP.

(8L)

Module IV:
Synchronization and Replication: Introduction - Clocks, events and process states - Synchronizing physical clocks- Logical time and logical clocks - Global states - Coordination and Agreement - Introduction - Distributed mutual exclusion - Elections - Transactions and Concurrency Control- Transactions -Nested transactions - Locks - Optimistic concurrency control - Timestamp ordering - Atomic Commit protocols -Distributed deadlocks - Replication - Case study - Coda.

(8L)

Module V:
Process Management: Process Migration: Features, Mechanism – Threads, Models, Issues, Implementation.
Resource Management: Introduction- Features of Scheduling Algorithms -Task Assignment Approach - Load Balancing Approach - Load Sharing Approach

(8L)

Books recommended:

TEXT BOOK

1. G. Coulouris, J. Dollimore, and T. Kindberg, "Distributed Systems Concepts and Design", 5th Edition, Pearson Education, 2012. **(T1)**

REFERENCE BOOK

1. A. S. Tanenbaum, M. Van Steen, "Distributed Systems: Principles and Paradigms", Pearson Education, 2007. **(R1)**
2. P. K. Sinha, "Distributed Systems: Concepts and Design", Prentice Hall, 2007. **(R2)**

Course Evaluation:

Individual assignment, Theory (Quiz and End semester) examinations

Gaps in the syllabus (to meet Industry/Profession requirements):

POs met through Gaps in the Syllabus:

Topics beyond syllabus/Advanced topics/Design:

POs met through Topics beyond syllabus/Advanced topics/Design: Teaching through research papers.

Course Delivery Methods

CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Assignments/Seminars
CD3	Laboratory experiments/teaching aids
CD4	Industrial/guest lectures
CD5	Industrial visits/in-plant training
CD6	Self- learning such as use of NPTEL materials and internets
CD7	Simulation

MAPPING BETWEEN COURSE OUTCOMES AND PROGRAM OUTCOMES

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	2	2	2	1	2
CO2	3	3	3	1	3	1
CO3	2	3	3	3	2	1
CO4	3	3	3	3	1	1
CO5	2	2	3	2	1	1

If satisfying and < 34% = L, 34-66% = M, > 66% = H

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,CD6,CD7
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD3,CD4,CD5,CD7

PROGRAM ELECTIVE - I

COURSE INFORMATION SHEET

Course code: CS506
Course title: MACHINE LEARNING
Pre-requisite(s): Design of Algorithms, Mathematics 2, Artificial Intelligence
Co- requisite(s):
Credits: L: 3 T: 0 P: 0
Class schedule per week: 03
Class: M. Tech
Semester / Level: I/5
Branch: Computer Science and Engineering

Course Objectives
This course enables the students:

1.	To formulate machine learning problems corresponding to different applications.
2.	To understand various supervised, semi-supervised and unsupervised machine learning algorithms.
3.	To familiarize various machine learning software libraries and data sets publicly available.
4.	To develop machine learning based system for various real-world problems.
5.	To assess how the choice of a machine-learning algorithm impacts the accuracy of a system.

Course Outcomes
After the completion of this course, students will be able to:

CO1	Formulate machine learning problems corresponding to different applications: data, model selection, model complexity
CO2	Demonstrate understanding of a range of machine learning algorithms along with their strengths and weaknesses
CO3	Implement machine learning solutions to classification, regression, and clustering problems
CO4	Design and implement various machine learning algorithms in a range of real-world applications
CO5	Evaluate and analyse the performance of a machine-learning algorithm or a system based on machine learning algorithm.

SYLLABUS

Module I:
Introduction to Machine Learning
Machine Learning – what and why? Basics of Linear Algebra and Statistics, Overview of target function representations; Linear Regression.

(8L)

Module II:
Supervised Learning
Basics of Feature Selection and Evaluation, Decision Tree, Overfitting and Pruning, Logistic regression, Support Vector Machine and Kernel; Noise, bias-variance trade-off, under-fitting and over-fitting concepts.

(8L)

Module III:
Neural Networks
Perceptrons: representational limitation and gradient descent training. Multilayer networks and backpropagation. Hidden layers and constructing intermediate, distributed representations. Overfitting, learning network structure, recurrent networks.

(8L)

Module IV:
Unsupervised and Semi Supervised Learning
Learning from unclassified data. Clustering. Hierarchical Agglomerative Clustering. K-means partitional clustering. Expectation maximization (EM) for soft clustering. Semi-supervised learning with EM using labeled and unlabled data.

(8L)

Module V:
Ensemble
Committees of multiple hypotheses, bagging, boosting, active learning with ensembles.

(8L)

Books recommended:

TEXT BOOK

1. Tom Mitchell, "Machine Learning", Latest Edition, Mc-Graw Hill. **(T1)**

REFERENCE BOOK

1. Shai Shalev-Shwartz, and Shai Ben-David, "Understanding Machine Learning", Cambridge University Press, 2017. **(R1)**
2. Christopher Bishop, "Pattern Recognition and Machine Learning", Springer, 2006. **(R2)**

Course Evaluation:
Individual assignment, Theory (Quiz and End semester) examinations

Gaps in the syllabus (to meet Industry/Profession requirements):

POs met through Gaps in the Syllabus:

Topics beyond syllabus/Advanced topics/Design:

POs met through Topics beyond syllabus/Advanced topics/Design: Teaching through research papers.

Course Delivery Methods

CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Assignments/Seminars
CD3	Laboratory experiments/teaching aids
CD4	Industrial/guest lectures
CD5	Industrial visits/in-plant training
CD6	Self- learning such as use of NPTEL materials and internets
CD7	Simulation

MAPPING BETWEEN COURSE OUTCOMES AND PROGRAM OUTCOMES

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	3	2	3	1	1
CO2	3	2	3	3	1	1
CO3	3	3	3	2	1	2
CO4	3	3	3	3	2	2
CO5	3	3	3	2	2	1

If satisfying and < 34% = L, 34-66% = M, > 66% = H

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,CD6,CD7
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD3,CD4,CD5,CD7

COURSE INFORMATION SHEET

Course code: IT503
Course title: WIRELESS SENSOR NETWORKS
Pre-requisite(s): Basic Networking Fundamentals
Co-requisite(s):
Credits: L: 3 T: 0 P: 0
Class schedule per week: 03
Class: M. Tech
Semester / Level: I/5
Branch: Computer Science and Engineering
Name of Teacher:

Course Objectives

This course enables the students to:

1.	Familiarize with the principles of sensor nodes, network deployment and architectures.
2.	Know the data transmission and routing protocols. Know the differences among different networks.
3.	Analyze or compare the performance of different routing and MAC protocol
4.	Evaluate the performance of different MAC protocols and clustering algorithm
5.	Compute the throughput and channel utilization for different network scenarios.

Course Outcomes

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

CO1	Obtain a broad understanding about the network architecture of wireless sensor network.
CO2	Understand all basic characteristics of wireless sensor networks and sensor nodes.
CO3	Understand the principles of data transmission, clustering algorithm and routing protocols.
CO4	Analyze and evaluate different constraint of wireless sensor network, e.g., coverage, power management, security and data collisions.
CO5	Design and development of new sensor network architecture.

SYLLABUS

Module I:
Fundamentals of Sensor Networks
Introduction to wireless sensor networks, Wireless Sensor nodes- Sensing and sensors- challenges and constraints - node architecture-sensing subsystem, processor subsystem communication interfaces- prototypes, Application of Wireless sensors.

(8L)

Module II:
Communication Characteristics and Deployment Mechanisms
Wireless Transmission Technology and Systems-Radio Technology Primer-Available Wireless Technologies - Hardware- Telosb, Micaz motes- Time Synchronization Clock and the Synchronization Problem - Basics of time Synchronization-Time synchronization protocols - Localization- Ranging Techniques- Range based Localization-Range Free Localization- Event driven Localization.

(8L)

Module III:
Mac Layer
Overview-Wireless Mac Protocols-Characteristics of MAC protocols in Sensor networks – Contention free MAC Protocols- characteristics- Traffic Adaptive Medium Access-Y-MAC, Low energy Adaptive Clustering - Contention based MAC Protocols, Power Aware Multi-Access with signalling, Sensor MAC-Timeout MAC-Data gathering MAC.

(8L)

Module IV:
Routing in Wireless Sensor Networks
Design Issues in WSN routing- Data Dissemination and Gathering-Routing Challenges in WSN - Flooding-Flat Based Routing – SAR, Directed Diffusion, Hierarchical Routing-LEACH, PEGASIS - Query Based Routing- Negotiation Based Routing Geographical Based Routing- Transport layer- Transport Protocol Design issues, Performance of Transport Control Protocols.

(8L)

Module V:
Middleware and Security Issues
WSN middleware Principles-Middleware Architecture-Existing middleware - operating systems for wireless sensor networks-performance and traffic management - Fundamentals of network security-challenges and attacks - Protocols and mechanisms for security.

(8L)

Books recommended:

TEXT BOOK

1. WaltenegusDargie, Christian Poellabauer, "Fundamentals of Wireless Sensor Networks, Theory and Practice", Wiley Series on wireless Communication and Mobile Computing, 2011. **(T1)**
2. Kazem Sohraby, Daniel manoli, "Wireless Sensor networks- Technology, Protocols and Applications", Wiley Inter Science Publications 2010. **(T2)**

REFERENCE BOOK

1. Bhaskar Krishnamachari, "Networking Wireless Sensors", Cambridge University Press, 2005. **(R1)**
2. C.S Raghavendra, Krishna M.Sivalingam, Taiebznati, "Wireless Sensor Networks", Springer Science 2004. **(R2)**

Course Evaluation:

Individual assignment, Theory (Quiz and End semester) examinations

Gaps in the syllabus (to meet Industry/Profession requirements):

POs met through Gaps in the Syllabus:

Topics beyond syllabus/Advanced topics/Design:

POs met through Topics beyond syllabus/Advanced topics/Design: Teaching through research papers.

Course Delivery Methods

CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Assignments/Seminars
CD3	Laboratory experiments/teaching aids
CD4	Industrial/guest lectures
CD5	Industrial visits/in-plant training
CD6	Self- learning such as use of NPTEL materials and internets
CD7	Simulation

MAPPING BETWEEN COURSE OUTCOMES AND PROGRAM OUTCOMES

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	2	1	1	1	1
CO2	3	2	1	1	1	1
CO3	3	3	1	1	1	1
CO4	2	3	2	1	1	1
CO5	2	3	3	2	1	1

If satisfying and < 34% = L, 34-66% = M, > 66% = H

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,CD6,CD7
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD3,CD4,CD5,CD7

COURSE INFORMATION SHEET

Course code: CS507
Course title: COMPUTABILITY AND COMPLEXITY THEORY
Pre-requisite(s): Automata Theory and Computer algorithms
Co-requisite(s):
Credits: L: 3 T: 0 P: 0
Class schedule per week: 03
Class: M. Tech
Semester / Level: I/5
Branch: Computer Science and Engineering
Name of Teacher:

Course Objectives
This course enables the students to:

1.	Give introduction to the mathematical foundations of computation including automata
2.	Learn about the issues in finite representations for languages and machines, as well as gain a more formal understanding of algorithms and procedures.
3.	Motivate and expose to the fundamental understanding of computation under resource constraints.
4.	Set a research level exposure to deeper topics in complexity theory.

Course Outcomes
After the completion of this course, students will be able to:

CO1	Relate formal languages and mathematical models of computation
CO2	Attain knowledge about different types of languages and the corresponding machines for computations
CO3	Understand the limitations on what computers can't do, and learn examples of unsolvable problems
CO4	Analyse P, NP, NP-C, NP-Hard, Tractable and Intractable problems
CO5	Explain reduction of problems for easy and hard problems

SYLLABUS

Module I:
Basic background on automata and languages, Types of automata and languages, Turing machines, Encoding and Enumeration of Turing Machines, k-tape Turing machines, non-deterministic Turing machines, Universal Turing machine, Resource bounded computation, Halting problem

(8L)

Module II:
Context Sensitive Language and Chomosky Hierarchy Recursive enumerable languages, Recursive languages, Decidable and recognizable language, Turing-decidable languages, Turing-recognizable languages, Kolmogorov Complexity

(8L)

Module III:
Primitive recursive function, partial recursive function, Recursive and recursive enumeration sets, Programming systems, Unsolvable problems, a non-recursive language and an unsolvable problem, Rice Theorem, More unsolvable problems, PCP

(8L)

Module IV:
Measuring complexity- Big Oh, small oh and other notations, Analysing algorithms, Time and space complexity of a Turing machine, Complexity analysis of multi-tape TM

(8L)

Module V:
Complexity classes: P, NP, NP-C, NP-Hard problem, PSPACE, NP-complete problems-clique, vertex cover, Hamiltonian cycle, graph colouring problem, graph isomorphism, Reduction from NP-C problem to another problem, Cook-Levin Theorem,Tractable and Intractable problems, Reducing one problem to another problem**,** Additional classes of problems- RP**,** ZPP

(8L)

Books recommended:

TEXT BOOK

1. Lewis H.R., Papadimitriou C.H., "Elements of the Theory of Computation", PHI Publ., 2nd edition, New Delhi. **(T1)**

REFERENCE BOOK

1. Hopcroft J.E., Motwani R. and Ullman J.D, "Introduction to Automata Theory, Languages and Computations", Second Edition, Pearson Education, 2008. **(R1)**
2. John Martin, "Introduction to Languages and the Theory of Computation", 3rd ed. McGraw Hill, New York, NY, 2003. **(R2)**
3. Dexter Kozen, "Theory of Computation", Springer publication. **(R3)**

Course Evaluation:
Individual assignment, Theory (Quiz and End semester) examinations

Gaps in the syllabus (to meet Industry/Profession requirements):

POs met through Gaps in the Syllabus:

Topics beyond syllabus/Advanced topics/Design:

POs met through Topics beyond syllabus/Advanced topics/Design: Teaching through research papers.

Course Delivery Methods

CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Assignments/Seminars
CD3	Laboratory experiments/teaching aids
CD4	Industrial/guest lectures
CD5	Industrial visits/in-plant training
CD6	Self- learning such as use of NPTEL materials and internets
CD7	Simulation

MAPPING BETWEEN COURSE OUTCOMES AND PROGRAM OUTCOMES

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	2	1	1	1	1
CO2	3	2	1	1	1	1
CO3	3	3	1	1	1	1
CO4	2	3	2	1	1	1
CO5	2	3	3	2	1	1

If satisfying and < 34% = L, 34-66% = M, > 66% = H

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,CD6,CD7
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD3,CD4,CD5,CD7

OPEN ELECTIVE – I

COURSE INFORMATION SHEET

Course code: CS514
Course title: SOFTWARE METRICS
Pre-requisite(s): Software Engineering, Software Testing
Co- requisite(s):
Credits: L: 3 T: 0 P: 0
Class schedule per week: 03
Class: M. Tech
Semester / Level: I/5
Branch: Computer Science and Engineering
Name of Teacher:

Course Objectives

After the completion of this course, students will be:

1.	Provide a basic understanding and knowledge of the software metrics.
2.	Understand the importance of Metrics data collection.
3.	Analysis and Metrics for object-oriented systems.
4.	Understand external product attributes, Dynamic Metrics and Resource measurement.

Course Outcomes

After the completion of this course, students will be:

CO1	Able to understand the importance of the software development process.
CO2	Analyze the importance of modelling and modelling language.
C03	Design and develop correct and robust software products.
CO4	Explain the business requirements pertaining to software development.
CO5	Design Metrics for object-oriented systems.

SYLLABUS

Module I:

Basics of measurement : Measurement in software engineering Scope of software metrics Representational theory of measurement Measurement and models Measurement scales Meaningfulness in measurement Goal-based framework for software measurement Classifying software measures Determining what to measure Software measurement validation Empirical investigation Types of investigation Planning and conducting investigations.

(8L)

Module II:

Measuring size, Aspects of software size Length, functionality and complexity Measuring structure Types of structural measures Control-flow structure Modularity and information flow attributes Data structures.

(8L)

Module III:

Modeling software quality Measuring aspects of software quality Software reliability Basics of software reliability Software reliability problem Parametric reliability growth models Predictive accuracy Recalibration of software-reliability growth predictions Importance of operational environment Wider aspects of software reliability.

(8L)

Module IV:

The intent of object-oriented metrics Distinguishing characteristics of object-oriented metrics Various object-oriented metric suites LK suite CK suite and MOOD metrics
Runtime Software Metrics Extent of Class Usage Dynamic Coupling Dynamic Cohesion and Data Structure Metrics.

(8L)

Module V:

The intent of component-based metrics, Distinguishing characteristics of comp.
Measuring productivity, teams, tools, and methods.

(8L)

Books recommended:

TEXT BOOK

1. "Software Metrics: A rigorous and Practical Approach" by Norman E. Fenton and Shari Lawrence Pfleeger, International Thomson Computer Press, 2nd Edition, 1997. **(T1)**
2. "Applied Software Measurement" by Capers Jones, McGraw Hill, 2008. **(T2)**

REFERENCE BOOK

1. "Object-Oriented Software Metrics" by Mark Lorenz, Jeff Kidd, Prentice Hall, 1994. **(R1)**
2. "Practical Software Metrics For Project Management And Process Improvement" by Robert B Grady, Hewlett Packard Professional Books, 1st Edition, 2004. **(R2)**

Course Evaluation:

Individual assignment, Theory (Quiz and End semester) examinations

Gaps in the syllabus (to meet Industry/Profession requirements):

POs met through Gaps in the Syllabus:

Topics beyond syllabus/Advanced topics/Design:

POs met through Topics beyond syllabus/Advanced topics/Design: Teaching through research papers.

Course Delivery Methods

CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Assignments/Seminars
CD3	Laboratory experiments/teaching aids
CD4	Industrial/guest lectures
CD5	Industrial visits/in-plant training
CD6	Self- learning such as use of NPTEL materials and internets
CD7	Simulation

MAPPING BETWEEN COURSE OUTCOMES AND PROGRAM OUTCOMES

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	3	2	2	1	1
CO2	2	2	3	1	1	1
CO3	1	3	3	2	2	2
CO4	3	3	3	2	1	1
CO5	2	1	3	3	3	1

If satisfying and < 34% = L, 34-66% = M, > 66% = H

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,CD6,CD7
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD3,CD4,CD5,CD7

COURSE INFORMATION SHEET

Course code: CS522
Course title: PATTERN RECOGNITION AND APPLICATION
Pre-requisite(s): Linear Algebra, Vector Calculus, Data Structure& Programming
Co- requisite(s):
Credits: L: 3 T: 0 P: 0
Class schedule per week: 03
Class: M. Tech
Semester / Level: I/5
Branch: Computer Science and Engineering
Name of Teacher:

Course Objectives

This course enables the students to:

1.	Be familiar with both the theoretical and practical aspects Pattern Recognition.
2.	Have described the foundation of pattern formation, measurement, and analysis.
3.	Understand the mathematical and computer aspects of while extracting features of an object.
4.	Learn the techniques of clustering and classification for various applications.

Course Outcomes

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

CO1	Apply their knowledge on Real World Problems while converting these problems to computer compatible problems for Pattern Recognition.
CO2	Solve Decision-making model using Statistical and Mathematical Decision Theory.
CO3	Design clusters for various Pattern using classical and Modern clustering techniques.
CO4	Analyzing various Techniques for Pattern Classification and Clustering.
CO5	Develop Model for Pattern classification through Probabilistic or fuzzy.

SYLLABUS

Module I:
Introduction: Feature Vectors, Classifiers, Supervised, Unsupervised, MATLAB Tools. Classifiers Based on Bayesian Theory, Linear Classifiers, Nonlinear Classifiers.

(8L)

Module II:
Feature Selection, Feature Generation I: Data Transformation and Dimensionality Reduction, Feature Generation II.

(8L)

Module III:
Template Matching, Context Dependent Classification, Super vised Learning.

(8L)

Module IV:
Clustering Basic Concepts, sequential Algorithms.

(8L)

Module V:
Hierarchical algorithms, Fuzzy clustering, probabilistic clustering, Hard Clustering, Optimization.

(8L)

Books recommended:

TEXT BOOK

1. "Pattern Recognition" by S Theodoridis, K Koutroumbas, Elsevier, 5th Edition, 2015. **(T1)**
2. "Pattern Recognition" by N Narshima Murty, Springer, University press, 2nd Edition, 2015. **(T2)**

REFERENCE BOOK

1. R. O. Duda et.al, "Pattern Classification", 2nd Edition, John Wiley, New York, 2002. **(R1)**

Course Evaluation:
Individual assignment, Theory (Quiz and End semester) examinations

Gaps in the syllabus (to meet Industry/Profession requirements):

POs met through Gaps in the Syllabus:

Topics beyond syllabus/Advanced topics/Design:

POs met through Topics beyond syllabus/Advanced topics/Design: Teaching through research papers.

Course Delivery Methods

CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Assignments/Seminars
CD3	Laboratory experiments/teaching aids
CD4	Industrial/guest lectures
CD5	Industrial visits/in-plant training
CD6	Self- learning such as use of NPTEL materials and internets
CD7	Simulation

MAPPING BETWEEN COURSE OUTCOMES AND PROGRAM OUTCOMES

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	3	3	3	2	1
CO2	3	3	3	2	1	1
CO3	3	2	2	3	1	1
CO4	3	3	2	1	3	1
CO5	2	1	2	2	1	1

If satisfying and < 34% = L, 34-66% = M, > 66% = H

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,CD6,CD7
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD3,CD4,CD5,CD7

COURSE INFORMATION SHEET

Course code: IT518
Course title: INTERNET OF THINGS
Pre-requisite(s):
Co-requisite(s):
Credits: L: 3 T: 0 P: 0
Class schedule per week: 03
Class: M. Tech
Semester / Level: I/5
Branch: Computer Science and Engineering
Name of Teacher:

Course Objectives

This course enables the students to:

1.	Understand the basic concept and the IoT Paradigm
2.	Know the state of art architecture for IoT applications
3.	Learn the available protocols used for IoT
4.	Design basic IoT Applications.
5.	Evaluate optimal IoT applications.

Course Outcomes

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

CO1	Identify the IoT Components and its capabilities
CO2	Explain the architectural view of IoT under real world constraints
CO3	Analyse the different Network and link layer protocols
CO4	Evaluate and choose among the transport layer protocols
CO5	Design an IoT application

SYLLABUS

Module I:
IoT-An Architectural Overview
Building an architecture, Main design principles and needed capabilities, An IoT architecture outline, standards considerations. M2M and IoT Technology Fundamentals- Devices and gateways, Local and wide area networking, Data management, Business processes in IoT, Everything as a Service (XaaS), M2M and IoT Analytics, Knowledge Management.

(8L)

Module II:
IoT Architecture-State of the Art
Introduction, State of the art, Reference Model and architecture, IoT reference Model - IoT Reference Architecture; Introduction, Functional View, Information View, Deployment and Operational View, Other Relevant architectural views. Real-World Design Constraints-Introduction, Technical Design constraints.

(8L)

Module III:
IoT Data Link Layer & Network Layer Protocols
PHY/MAC Layer (3GPP MTC, IEEE 802.11, IEEE 802.15), WirelessHART, Z-Wave, Bluetooth Low Energy, Zigbee Smart Energy, DASH7 - Network Layer-IPv4, IPv6, 6LoWPAN, 6TiSCH,ND, DHCP, ICMP, RPL, CORPL, CARP.

(8L)

Module IV:
Transport & Session Layer Protocols
Transport Layer (TCP, MPTCP, UDP, DCCP, SCTP)-(TLS, DTLS) – Session Layer-HTTP, CoAP, XMPP, AMQP, MQTT.

(8L)

Module V:
Layer Protocols & Security
Service Layer -oneM2M, ETSI M2M, OMA, BBF – Security in IoT Protocols – MAC 802.15.4, 6LoWPAN, RPL, Application Layer protocols.

(8L)

Books recommended:

TEXT BOOK

1. Jan Holler, VlasiosTsiatsis, Catherine Mulligan, Stefan Avesand, StamatisKarnouskos, David Boyle, "From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence", 1st Edition, Academic Press, 2014. **(T1)**
2. Peter Waher, "Learning Internet of Things", PACKT publishing, BIRMINGHAM – MUMBAI. **(T2)**

REFERENCE BOOK

1. Bernd Scholz-Reiter, Florian Michahelles, "Architecting the Internet of Things", ISBN 978-3-642-19156-5 e-ISBN 978-3-642-19157-2, Springer. **(R1)**
2. Daniel Minoli, "Building the Internet of Things with IPv6 and MIPv6". **(R2)**

Course Evaluation:
Individual assignment, Theory (Quiz and End semester) examinations

Gaps in the syllabus (to meet Industry/Profession requirements):

POs met through Gaps in the Syllabus:

Topics beyond syllabus/Advanced topics/Design:

POs met through Topics beyond syllabus/Advanced topics/Design: Teaching through research papers.

Course Delivery Methods

CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Assignments/Seminars
CD3	Laboratory experiments/teaching aids
CD4	Industrial/guest lectures
CD5	Industrial visits/in-plant training
CD6	Self- learning such as use of NPTEL materials and internets
CD7	Simulation

MAPPING BETWEEN COURSE OUTCOMES AND PROGRAM OUTCOMES

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	3	3	3	2	1
CO2	3	3	3	2	1	1
CO3	3	2	2	3	1	1
CO4	3	3	2	1	3	1
CO5	2	1	2	2	1	1

If satisfying and < 34% = L, 34-66% = M, > 66% = H

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,CD6,CD7
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD3,CD4,CD5,CD7

COURSE INFORMATION SHEET

Course code: IT522
Course title: CYBER SECURITY AND DIGITAL FORENSICS
Pre-requisite(s): Computer Basics, Programming and Problem solving
Co-requisite(s):
Credits: L: 3 T: 0 P: 0
Class schedule per week: 03
Class: M. Tech.
Semester / Level: I/5
Branch: Computer Science and Engineering
Name of Teacher:

Course Objectives
This course enables the students:

1.	Know about computer-based crime.
2.	Understand Technical and legal aspects of computer crime investigations
3.	Know the limitations of law and its enforcement agencies.
4.	Learn the procedures of recovering computer evidence and seize process.

Course Outcomes
After the completion of this course, students will be able to:

CO1	Exposure on computer-based crime.
CO2	Technical and legal aspects of computer crime investigations
C03	Know the limitations of law and its enforcement agencies.
CO4	Learn the procedures of recovering computer evidence and seize process.
CO5	Apply techniques for finding, preserving, presenting, and extracting information from the digital devices.

SYLLABUS

Module I:
Introduction: Cyberspace and Criminal Behavior, Traditional problems associated with computer-based crime, e cash problems, Computer Technology and History: Computer Language, Hardware, software, operating system, Internet, Network language.

(8L)

Module II:
Early Hackers and Theft Components: Phreakers, Hacking, Commodities, Intellectual property.
Contempory computer crime: web based criminal activity, money laundering,

(8L)

Module III:
Identity theft and identity fraud: Typologies of internet theft, virtual identity, credit identity. Prevalence and victimology, physical methods, of identity theft, phishing, spyware, trojans, insurance and loan fraud, immigration fraud. Terrorism and organized crime:Terror online, criminal activities, organized crime as cyber gangs., technology used in organized crime. Data piracy.

(8L)

Module IV:
Avenues for Prosecution and Government efforts: Act, Law enforcement agencies, International efforts, Cyber law and its amendments of current state, other legal considerations.

(8L)

Module V:
Forensic Terminologies and Developing forensic capabilities, Searching and seizing computer related evidence, Processing of evidence and report preparation.

(8L)

Books recommended:

TEXT BOOK

1. "Computer Forensics and Cyber Crime" by M. T. Britz, Pearson Education, First Impression, 2012. **(T1)**
2. "Computer Crime and investigation" by E Casey, Springer, 1st Edition, 2001. **(T2)**

REFERENCE BOOK

1. "Computer Crime Investigations and Law" by C Easttom & D. J. Taylor, Carenage Learning. **(R1)**

Course Evaluation:
Individual assignment, Theory (Quiz and End semester) examinations

Gaps in the syllabus (to meet Industry/Profession requirements):

POs met through Gaps in the Syllabus:

Topics beyond syllabus/Advanced topics/Design:

POs met through Topics beyond syllabus/Advanced topics/Design: Teaching through research papers.

Course Delivery Methods

CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Assignments/Seminars
CD3	Laboratory experiments/teaching aids
CD4	Industrial/guest lectures
CD5	Industrial visits/in-plant training
CD6	Self- learning such as use of NPTEL materials and internets
CD7	Simulation

MAPPING BETWEEN COURSE OUTCOMES AND PROGRAM OUTCOMES

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	3	1	1	2	1
CO2	3	3	3	1	1	1
CO3	3	3	3	2	1	1
CO4	2	3	2	2	2	1
CO5	2	3	2	2	2	1

If satisfying and < 34% = L, 34-66% = M, > 66% = H

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,CD6,CD7
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD3,CD4,CD5,CD7

COURSE INFORMATION SHEET

Course code: CS503
Course title: ADVANCED DATA STRUCTURES LAB
Pre-requisite(s):
Co- requisite(s): Advanced Data Structures
Credits: L: 0 T: 0 P: 2
Class schedule per week: 04
Class: M. Tech
Semester / Level: I/5
Branch: Computer Science and Engineering
Name of Teacher:

Course Objectives

This course enables the students to:

1.	The student should be able to choose appropriate data structures, understand the ADT/libraries, and use it to design algorithms for a specific problem.
2.	Students should be able to understand the necessary mathematical abstraction to solve problems.
3.	To familiarize students with advanced paradigms and data structure used to solve algorithmic problems.
4.	Student should be able to come up with analysis of efficiency and proofs of correctness.

Course Outcomes

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

CO1	Understand to implement the symbol table using hashing techniques.
CO2	Develop program for AVL, Red-Black trees, B-trees and Splay trees
CO3	Develop program for text processing applications
CO4	Learn the basic working of advanced heaps
CO5	Understand the implementation of symbol table using hashing techniques

SYLLABUS

List of Programs as Assignments:

1. **Lab Assignment No: 1**
   Write a program to implement a dictionary using the following ADTs. We assume all the entries in the dictionary to be distinct integers.

   a) Binary Search Tree (BST)
   b) Red Black Tree (RBT)

   Each ADT should support five operations, void Insert(val), boolean Delete(val), boolean Search(val), void ClearADT() and void DisplayADT(). Both search and delete operations should respond with a boolean value indicating whether the search/delete was successful or not. The objective of this assignment is to compare the performance of BST and RBT ADTs. You have to compute the time taken for completion of operations and study how the running times of ADT operations will vary across the two ADT implementations.

2. **Lab Assignment No: 2**
   Implement data structures to maintain a list of elements. In particular, implement the list as an array and as a linked list. Write a program to remove duplicates from the list. The code for remove duplicates functionality should remain the same across the two implementation of the list.
   Implement the following sequence of operations on the skip list :
   a) remove() b) insert() c) Search() d) Update()
   Assume the coin flips for the first insertion yield two heads followed by tails, and those for the second insertion yield three heads followed by tails.

3. **Lab Assignment No: 3**
   Implement different Hashing functions and Collision Resolution Techniques.

4. **Lab Assignment No: 4**
   Implement Kruskal's algorithm for finding the minimum spanning tree of a given (positively) weighted (undirected) graph G. You must use the Union-Find data structure that implements both "union by rank" and "path compression" heuristics.

5. **Lab Assignment No: 5 &6**
   Implement the following priority queues:
   a) Binary Heap b) Binomial Heap c) Fibonacci Heap

   Your program should then create an appropriate priority queue object and perform makeHeap() method. After that, you should give the user menu options to insert(key), delete(key), extractMin(), findMin(), decrease(key), increase(key) updateKey(currentKey, newKey), and displayHeap(filename). You may assume that keys will be unique. For displayHeap(filename), you must output the tree structure of the priority queue (including linked list edges in Binomial Heap, etc) as a directed tree in the dot language format1 and store it in a file named filename (given as parameter).

6. **Lab Assignment No: 7&8**
   Implement Text Processing using Huffman Coding.

   Implement the compact representation of the suffix trie for the string "minimize minime".

   Implement a standard trie for the following set of strings: { abab, baba, ccccc, bbaaaa, caa, bbaacc, cbcc, cbca }.

7. **Lab Assignment No: 9 &10**
   One way to mask a message, *M*, using a version of *steganography*, is to insert random characters into *M* at pseudo-random locations so as to expand *M* into a larger string, *C*.

   For instance, the message,

   ILOVEMOM,

   could be expanded into

   AMIJLONDPVGEMRPIOM.

   It is an example of hiding the string, *M*, in plain sight, since the characters in *M* and *C* are not encrypted. As long as someone knows where the random

Books recommended:

TEXT BOOK

1. Mark Allen Weiss, "Data Structures and Algorithm Analysis in C++," 2nd Edition, Pearson, 2004. **(T1)**
2. Peter Brass, "Advanced Data Structures," Cambridge University Press, 1st Edition. **(T2)**
3. M T Goodrich, & Roberto Tamassia, "Algorithm Design," John Wiley, 2002. **(T3)**

Course Evaluation:
Day to day progressive evaluation, Lab Quizzes, Surprise Tests, Online Lab performance and Viva Voce

Gaps in the syllabus (to meet Industry/Profession requirements):
Implementing of real world problems
POs met through Gaps in the Syllabus:

Topics beyond syllabus/Advanced topics/Design:

POs met through Topics beyond syllabus/Advanced topics/Design: Teaching through research papers.

Course Delivery Methods

CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Assignments/Seminars
CD3	Laboratory experiments/teaching aids
CD4	Industrial/guest lectures
CD5	Industrial visits/in-plant training
CD6	Self- learning such as use of NPTEL materials and internets
CD7	Simulation

MAPPING BETWEEN COURSE OUTCOMES AND PROGRAM OUTCOMES

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	1	1	1	1	1
CO2	3	2	2	1	1	1
CO3	2	3	3	2	1	1
CO4	3	1	1	1	1	1
CO5	3	1	1	1	1	1

If satisfying and < 34% = 1, 34-66% = 2, > 66% = 3

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD7

PROGRAM ELECTIVE LAB –I

COURSE INFORMATION SHEET

Course code: IT509
Course title: MATLAB PROGRAMMING
Pre-requisite(s):
Co- requisite(s):
Credits: L: 0 T: 0 P: 2
Class schedule per week: 04
Class: M. Tech
Semester / Level: I/5
Branch: Computer Science and Engineering
Name of Teacher:

Course Objectives

This course enables the students to:

1.	To familiarize the student in introducing and exploring MATLAB
2.	Develop mathematical thinking and problem-solving skill
3.	To enable the student on how to approach for solving Engineering problems using simulation tools.
4.	To provide a foundation in use of this software for real time app

Course Outcomes

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

CO1	Express programming & simulation for engineering problems.
CO2	Find importance of this software for Lab Experimentation.
CO3	Analyze problems and write basic mathematical ,electrical ,electronic problems in Matlab
CO4	Implement programming files with GUI Simulink.
CO5	Simulate basic Engineering problems

SYLLABUS

List of Programs as Assignments:

1. **Lab Assignment No: 1**
   Objective: To Understand and Implement Matrix Algebra
   1. To create Sparse matrices using the function sparse.
   2. To convert a sparse matrix to full matrix.

2. **Lab Assignment No: 2**
   Objective: To Understand and Implement Data Analysis
   1. To measure the daily high temperatures in three cities with different color.
   2. To solve a different cities temperature do the filter.

3. **Lab Assignment No: 3**
   Objective: To Understand and Implement Data Interpolation
   1. To Draw 2-D random data.
   2. To Draw Threshold of Human Hearing.

4. **Lab Assignment No: 4**
   Objective: To Understand and Implement Cubic Splines
   1. To design Spline differentiation and Integration.
   2. To design interpolated Spiral Y=f(X).

5. **Lab Assignment No: 5**
   Objective: To Understand and Implement Fourier Analysis
   1. To solve the use of the FFT, consider the problem of estimating the continuous Fourier transform of the signal
   $f(t)=2e^{-3t}$ $t \geq 0$, where f(t) is gven by
   $F(\omega)=2/(3+j\,\omega)$
   2. To design sawtooth Waveform at arbitrary points.

6. **Lab Assignment No: 6**
   Objective: To Understand and Implement Optimization
   1. To solve 1-D minimization and maximization.
   2. To design Rosenbrock's banana function.

7. **Lab Assignment No: 7**
   Objective: To Understand and Implement Differential Equations
   1. Design a van der Pol Solution.
   2. To solve Jacobian matrix.

8. **Lab Assignment No: 8**
   Objective: To Understand and Implement Two-Dimensional Graphics
   1. To add new plots to an existing plot by using the hold command.
   2. To create new Figure windows, use the figure command in the Command window or the **New Figure** selection from the **File** menu in the Command or Figure window.

9. **Lab Assignment No: 9**

Objective: To Understand and Implement Three- Dimensional Graphics

1. Plot = 2 with 0 5in polar coordinates.
2. Design a Surface plot using the surf function.

10. **Lab Assignment No: 10**

Objective: To Understand and Implement Images, Movies, and Sound

1. To display 8-bit intensity and RGB images.
2. To convert between indexed images and movie frames.

11. **Lab Assignment No: 11**

Objective: To Understand and Implement Graphical User Interfaces

1. Using uigetfile to find the startup.m file on the author's computer.
2. Write a program using the function **guidata,** which stores and retrieves data in the GUI figure **'ApplicationData'** property.
3. How to put walls by clicking a mouse button to make an arbitrary maze.
4. Design a GUI System.

Books recommended:

TEXT BOOK

1. Duane Hanselman, bruce Littlefield, Mastering MATLAB 7, Reason edu., 2nd edition, 2008. **(T1)**
2. Sandeep Nagar, Introduction to MATLAB for Engineers and Scientists: Solutions for Numerical Computation and Modeling, A Press, 2017. **(T2)**

REFERENCE BOOK

1. MATLAB Primer by MATHWORKS. **(R1)**

Course Evaluation:

Day to day progressive evaluation, Lab Quizzes, Surprise Tests, Online Lab performance and Viva Voce

Gaps in the syllabus (to meet Industry/Profession requirements):

Implementing of real world problems

POs met through Gaps in the Syllabus:

Topics beyond syllabus/Advanced topics/Design:

POs met through Topics beyond syllabus/Advanced topics/Design: Teaching through research papers.

Course Delivery Methods

CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Assignments/Seminars
CD3	Laboratory experiments/teaching aids
CD4	Industrial/guest lectures
CD5	Industrial visits/in-plant training
CD6	Self- learning such as use of NPTEL materials and internets
CD7	Simulation

MAPPING BETWEEN COURSE OUTCOMES AND PROGRAM OUTCOMES

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	3	2	2	1	1
CO2	3	3	2	2	1	1
CO3	2	3	2	1	2	1
CO4	3	3	2	2	1	1
CO5	3	2	1	1	1	1

If satisfying and < 34% = 1, 34-66% = 2, > 66% = 3

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD7

COURSE INFORMATION SHEET

Course code: IT510
Course title: JAVA PROGRAMMING
Pre-requisite(s):
Co- requisite(s): None
Credits: L: 0 T: 0 P: 2
Class schedule per week: 04
Class: M. Tech
Semester / Level: I/5
Branch: Computer Science and Engineering
Name of Teacher:

Course Objectives

This course enables the students to:

1.	To familiarize the student in introducing and exploring JAVA.
2.	Knowledge of the structure and model of the Java programming language.
3.	Use the Java programming language for various programming technologies.
4.	To provide a foundation for Java programming language to solve the given problems.

Course Outcomes

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

CO1	Write, compile, and execute Java programs that may include basic data types and control flow constructs using J2SE or other Integrated Development Environments (IDEs)
CO2	Write, compile, and execute Java programs manipulating Strings and text documents.
CO3	Write, compile, execute Java programs that include GUIs and event driven programming
CO4	Create Applets and GUI
CO5	Executing Client server and socket programming

SYLLABUS

List of Programs as Assignments:

1. **Lab Assignment No: 1**
   Objective: To Understand and Implement COLLECTION FRAMEWORK
   1. To create sparse matrices using the function sparse.
   2. To convert a sparse matrix to full matrix.
2. **Lab Assignment No: 2**
   Objective: To Understand and Implement Generic Programming
   1. Write the tasks performed by type erasure?
   2. Write a generic method to exchange the positions of two different elements in an array?
3. **Lab Assignment No: 3**
   Objective: To Understand and Implement REFLECTION
   1. Write a program that finds and displays inheritance hierarchy of a specified class?
   2. Write a program that shows all public fields of a specified class?
4. **Lab Assignment No: 4**
   Objective: GUI Development with Swing
   1. Working with Text Fields
   2. Working with Buttons
   3. Working with Lists
   4. Working with Scroll Panes
5. **Lab Assignment No: 5**
   Objective: Implementing Robust Geometric Primitives
   1. Java Program to Apply Above-Below-on Test to Find the Position of a Point with respect to a Line
   2. Java Program to Compute the Area of a Triangle Using Determinants
   3. Java Program to Compute the Volume of a Tetrahedron Using Determinants
   4. Java Program to Find the Area of any Polygon Using Triangulation
6. **Lab Assignment No: 6**
   Objective: To Understand and Implement examples on "Convex Hull"
   1. Java Program to Implement Graham Scan Algorithm to Find the Convex Hull
   2. Java Program to Implement Gift Wrapping Algorithm in Two Dimensions
   3. Java Program to Implement Jarvis March to Find the Convex Hull
7. **Lab Assignment No: 7**
   Objective: To Understand and Implement examples on "Nearest Neighbor Search"
   1. Java Program to Find the Nearest Neighbour Using K-D Tree Search
   2. Java Program to Find Nearest Neighbour Using Voronoi Diagram
8. **Lab Assignment No: 8**
   Objective: To Understand and Implement Network Programming
   1. Working with URLs
   2. Socket Server Programming
   3. Client Server Programming
9. **Lab Assignment No: 9**
   Objective: To Understand and Implement SOCKET PROGRAMMING
   1.Write a java socket program to get the resource http://www.google.com/index.html using HTTP protocol?
   2. Write a program how do you get the IP address of a machine from its hostname?

10. **Lab Assignment No: 10**
    Objective: To Understand and Implement Java Web Applications
    1. Simple Servlet
    2. Java Web Applications - Get Request
    3. Java Web Applications - Post Request
11. **Lab Assignment No: 11**
    Objective: To Understand and Implement Advanced Java Input/output (NIO)
    1. File Copying With NIO
    2. Working with Buffers
    3. Working with File Data

Books recommended:

TEXT BOOKS

1. Uttam K Roy, Advanced JAVA Programming, Oxford University Press, 1st Edition , 2015 **(T1)**
2. Heerbert Schildt, Java: A Beginner's Guide, Seventh Edition,Oracle Press, 2014 **(T2)**

Course Evaluation:
Day to day progressive evaluation, Lab Quizzes, Surprise Tests, Online Lab performance and Viva Voce

Gaps in the syllabus (to meet Industry/Profession requirements):
Implementing of real world problems
POs met through Gaps in the Syllabus

Topics beyond syllabus/Advanced topics/Design:

POs met through Topics beyond syllabus/Advanced topics/Design: Teaching through research papers.

Course Delivery Methods

CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Assignments/Seminars
CD3	Laboratory experiments/teaching aids
CD4	Industrial/guest lectures
CD5	Industrial visits/in-plant training
CD6	Self- learning such as use of NPTEL materials and internets
CD7	Simulation

MAPPING BETWEEN COURSE OUTCOMES AND PROGRAM OUTCOMES

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	3	2	2	1	1
CO2	3	3	2	2	1	1
CO3	3	2	2	2	2	1
CO4	3	2	3	2	1	1
CO5	3	2	1	1	1	1

If satisfying and < 34% = 1, 34-66% = 2, > 66% = 3

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD7

COURSE INFORMATION SHEET

Course code: IT511
Course title: R PROGRAMMING
Pre-requisite(s):
Co- requisite(s): None
Credits: L: 0 T: 0 P: 2
Class schedule per week: 04
Class: M. Tech
Semester / Level: I/5
Branch: Computer Science and Engineering
Name of Teacher:

Course Objectives
This course enables the students to:

1.	To familiarize the student in introducing and exploring R
2.	Develop basic thinking for data analysis.
3.	To enable the student on how to approach for statistical Analysis
4.	To provide a foundation in use of this software

Course Outcomes
After the completion of this course, students will be able to:

CO1	Manipulate data within R
CO2	Perform basic data analysis procedures
CO3	Create plots
CO4	Implement programming features like accessing R packages, writing R functions
CO5	Simulation & Profiling with R

SYLLABUS

List of Programs as Assignments:

1. Lab Assignment No: 1

Objective: To Understand and Implement Data Types

1. Data Types - R Objects and Attributes
2. Data Types - Vectors and Lists
3. Data Types - Matrices
4. Data Types - Factors
5. Data Types - Missing Values
6. Data Types - Data Frames
7. Data Types - Names Attribute
8. Data Types - Summary

2. Lab Assignment No: 2

Objective: To Understand and Implement Data Analysis

1. Reading Tabular Data
2. Reading Large Tables
3. Textual Data Formats
4. Connections: Interfaces to the Outside World
5. Subsetting - Basics
6. Subsetting - Lists
7. Subsetting - Matrices
8. Subsetting - Partial Matching
9. Subsetting - Removing Missing Values
10. Vectorized Operations

3. Lab Assignment No: 3

Objective: To Understand and Implement Swirl

1. Workspace and Files
2. Sequences of Numbers
3. Vectors

4. Lab Assignment No: 4

Objective: To Understand and Implement Cubic Splines

1. To design Spline differentiation and Integration.
2. To design interpolated Spiral Y=f(X).

5. Lab Assignment No: 5

Objective: To Understand and Implement Control Structures

1. If-else
2. Control Structures - For loops
3. Control Structures - While loops
4. Control Structures - Repeat, Next, Break

6. **Lab Assignment No: 6**

   Objective: To Understand and Implement Functions

   1. Functions (part 2)
   2. Scoping Rules - Symbol Binding
   3. Scoping Rules - R Scoping Rules
   4. Scoping Rules - Optimization Example (OPTIONAL)

7. **Lab Assignment No: 7 &8**

   Objective: To Understand and Implement Loop Functions and Debugging

   1. Loop Functions - lapply
   2. Loop Functions - apply
   3. Loop Functions - mapply
   4. Loop Functions - tapply
   5. Loop Functions – split

8. **Lab Assignment No: 9 & 10**

   Objective: To Understand and Implement Two-Dimensional Graphics

   1. Generating Random Numbers
   2. Simulation - Simulating a Linear Model
   3. Simulation - Random Sampling

Books recommended:

TEXT BOOKS

1. Norman Matloff, The Art of R Programming, A Tour of Statistical Software Design 1st Edition, **(T1)**
2. Hadley Wickham, Garrett Grolemund, R for Data Science: Import, Tidy, Transform, Visualize, and Model Data, Orielly, 1st Edition. **(T2)**

Course Evaluation:

Day to day progressive evaluation, Lab Quizzes, Surprise Tests, Online Lab performance and Viva Voce

Gaps in the syllabus (to meet Industry/Profession requirements):

Implementing of real world problems

POs met through Gaps in the Syllabus

Topics beyond syllabus/Advanced topics/Design:

POs met through Topics beyond syllabus/Advanced topics/Design: Teaching through research papers.

Course Delivery Methods

CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Assignments/Seminars
CD3	Laboratory experiments/teaching aids
CD4	Industrial/guest lectures
CD5	Industrial visits/in-plant training
CD6	Self- learning such as use of NPTEL materials and internets
CD7	Simulation

MAPPING BETWEEN COURSE OUTCOMES AND PROGRAM OUTCOMES

CO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	3	2	2	1	1
CO2	3	3	2	2	1	1
CO3	2	2	2	1	1	1
CO4	3	1	1	1	1	1
CO5	3	2	1	1	1	1

If satisfying and < 34% = 1, 34-66% = 2, > 66% = 3

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD7