VEERMATA JIJABAI TECHNOLOGICAL INSTITUTE
(VJTI) MATUNGA, MUMBAI 400 019
(Autonomous Institute affiliated to University of Mumbai)

Curriculum

(Scheme of Instruction & Evaluation and Course contents)
For
Post Graduate Programme Leading to

Masters of Engineering Degree
in
Computer Engineering

Department of Computer Engineering and Information Technology

Implemented from the batch admitted in Academic Year 2022-23

2024

M Tech Computer Engineering
Programme Outcomes

LIST OF PROGRAM OUTCOMES	
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 field of computer engineering and specialized topics in various domains of computer engineering.
PO4	An ability to apply mathematical modeling, algorithms and techniques in respective areas of computer engineering to solve complex engineering problems.
PO5	An ability to design and develop robust, reliable, scalable tools and techniques for knowledge-based systems to enhance lifelong learning.

MTech Computer Engineering
Scheme of Instruction and Evaluation
SEMESTER I

Scheme of Instruction					Scheme of Evaluation			
S.No	Course Code	Course Title	L-T-P	Credits	TA	MST	ESE	ESE hours
1	COCE5001S	Computational Methods (PSM)	3-0-0	3	20	20	60	3
2	COCE5011T	Advanced Algorithms (Core-1)	3-1-0	4	20	20	60	3
3	COCE5012S	Cloud Computing (Core-2)	3-0-0	3	20	20	60	3
4	COCE5021T	Program Elective -1	3-1-0	4	20	20	60	3
5	COCE5031S	Program Elective - 2	3-0-0	3	20	20	60	3
6	COCE5061S	Open Elective - 1	3-0-0	3	20	20	60	3
7	COCE5071L	Cloud Computing (Laboratory -1)	0-0-2	1	60%CIE		40	-
8	COCE5072L	Software Testing (Laboratory -3)	0-0-2	1	60%CIE		40	-
9	COCE5073L	Cross-Platform App Development (Laboratory -3)	0-0-2	1	60%CIE		40	-
10	MTEC081L	Liberal Learning-1	0-0-2	1	100%CIE		–	-
			28	24				

Abbreviations: **S**: Standalone, **T**: Standalone with Tutorial, **L**: Laboratory **TA**: Teachers Assessment / Term work Assessment, **MST**: Mid Semester Test, **ESE**: End Semester written Examination, **CIE**: Continuous in-Semester Evaluation

Program Elective -1			Program Elective - 2		
S.No	Course Code	Course Title	S.No	Course Code	Course Title
1.	COCE5021T	Computer Network Design	1.	COCE5031S	Software Engineering
2.	COCE5022T	Distributed Systems	2.	COCE5032S	Internet of Things
3.	COCE5023T	Computer Systems Performance Analysis	3.	COCE5033S	GPU Architecture and programming
4.	COCE5024T	Social Network Analysis	4.	COCE5034S	Graph Mining

Open Elective - 1		
S.No	Course Code	Course Title
1.	COCE5061S	Database Management Systems
2.	COCE5062S	Data Structure
3.	COCE5063S	Data Science

MTech Computer Engineering
Scheme of Instruction and Evaluation
SEMESTER II

Scheme of Instruction					Scheme of Evaluation			
S.No	Course Code	Course Title	L-T-P	Credits	TA	IST	ESE	ESE hours
1	COCE5002S	Research Methodology and IPR (Mandatory Learning)	3-0-0	3	20	20	60	3
2	COCE5013T	Big Data Analytics (Core-3)	3-1-0	4	20	20	60	3
3	COCE5014S	Soft computing (Core-4)	3-0-0	3	20	20	60	3
4	COCE5041T	Program Elective -3	3-1-0	4	20	20	60	3
5	COCE5051S	Program Elective - 4	3-0-0	3	20	20	60	3
6	COCE5062S	Open Elective - 2	3-0-0	3	20	20	60	3
7	COCE5074L	Devops (Laboratory-4)	0-0-2	1	60%CIE		40	-
8	COCE5075L	Deep Learning (Laboratory-5)	0-0-2	1	60%CIE		40	-
9	COCE5076L	Big Data Analytics (Laboratory -6)	0-0-2	1	60%CIE		40	-
10	MTEC082L	Liberal Learning-2	0-0-2	1	100%CIE		–	-
			28	24				

Abbreviations: **S**: Standalone, **T**: Standalone with Tutorial, **L**: Laboratory **TA**: Teachers Assessment / Term work Assessment, **MST**: Mid Semester Test, **ESE**: End Semester written Examination, **CIE**: Continuous in-Semester Evaluation

Program Elective -3			Program Elective - 4		
S.No	Course Code	Course Title	S.No	Course Code	Course Title
1.	COCE5041T	Network administration	1.	COCE5051S	Network Security
2.	COCE5042T	Ad-hoc Network	2.	COCE5052S	Blockchain Technology
3.	COCE5043T	Software Project Management	3.	COCE5053S	Parallel Algorithms
4.	COCE5044T	Software Architecture	4.	COCE5054S	Software Defined Network

Open Elective - 2		
S.No	Course Code	Course Title
1.	COCE5065S	Human Computer Interaction
2.	COCE5066S	Machine Learning

MTech Computer Engineering
Scheme of Instruction and Evaluation
SEMESTER III

Scheme of Instruction					Scheme of Evaluation
S.No	Course Code	Course Title	L-T-P	Credits	
1	COCE5091D	Skill Based Course (Project Stage -I)	—	5	100%CIE
2	COCE5092D	Skill Based Course (Project Stage -II)	—	5	100%CIE
3	COCE5101S	Self-Learning Course -1	—	1	100% ESE of 3 hours or credit transfer
4	COCE5201S	Self-Learning Course -2	—	1	100% ESE of 3 hours or credit transfer
5	COCE5201MNC	Mandatory Non-Credit Course	—	0	100% ESE of 3 hours or credit transfer
				12	

Scheme of Instruction and Evaluation
SEMESTER IV

Scheme of Instruction					Scheme of Evaluation
S.No	Course Code	Course Title	L-T-P	Credits	
1	COCE5093D	Skill Based Course (Project Stage -III)	—	5	100%CIE
2	COCE5093D	Skill Based Course (Project Stage -IV)	—	7	100%CIE
				12	

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5001S
Course Title	Computational Methods
Course Type	Program Specific Mathematics
Prerequisites: Fundamental of computer science, discrete mathematics and probability theory and statistics.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Analyze mathematical models and methods using proofs. **CO2.** Apply number theory principles for real world problems. **CO3.** Demonstrate counting usage in computer applications. **CO4.** Illustrate probability principles for addressing randomness in the applications. **CO5.** Practice recurrence in real life problems. **CO6.** Design advanced graph theoretic algorithms.	

	Course Contents	Hrs.	CO
1.	**Introduction to proofs:** Propositions, Predicates, Axiomatic Method, Proof by Cases, Proof by Contradiction, Well Ordering Proofs, Propositional Logic, Equivalence and Validity, Algebra of Propositions, Proof by Induction	8	CO1
2.	**Number Theory:** Divisibility, Modular Arithmetic, Multiplicative Inverses, Euler's Theorem, RSA Public Key Encryption	6	CO2
3.	**Counting:** Sums and Asymptotics, Counting Sequences, Counting Subsets, Pigeonhole Principle, Inclusion-Exclusion, Combinatorial Proofs, Generating Functions	6	CO3
4.	**Probability:** Events and Probability Spaces, Conditional Probability, Random Variables, Random Walks	8	CO4
5.	**Recurrences:** Linear Recurrences, Divide-and-Conquer Recurrences	6	CO5
6.	**Directed graphs and Partial Orders:** Vertex Degrees, Walks and Paths, Directed Acyclic Graphs and Scheduling, Partial Orders, Equivalence Relations	6	CO6

Text Books	
1.	Kolman, Bernard, Robert C. Busby, and Sharon Ross. Discrete mathematical structures. Prentice-Hall, Inc., 1995.
2.	Cormen, Thomas H., Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. Introduction to algorithms. MIT press, 2022.
Reference Books	
1.	Lehman, Eric, Tom Leighton, and Albert R. Meyer. Mathematics for computer science. Technical report, 2006. Lecture notes, 2010.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5011T
Course Title	Advanced Algorithms
Course Type	Core Subject

Prerequisites: Fundamental of computer science, discrete mathematics and probability theory and statistics.

Course Outcomes: At the end of the course student will be able to:
CO1. Analyze algorithms and determine efficiency of algorithm.
CO2. Design analysis algorithms using the greedy, dynamic programming, and divide and conquer techniques.
CO3. Design and build solutions for a real world problem using graphs.
CO4. Prove problems of P, NP, or NP-Complete.
CO5. Demonstrate geometric algorithms usage in real life problems.
CO6. Illustrate advanced algorithms techniques for NP Complete problems.

	Course Contents	Hrs.	CO
1.	**Introduction:** Asymptotic notation, recurrences, amortized analysis	6	CO1
2.	**Algorithm design techniques:** Greedy algorithms, divide-and-conquer algorithms, dynamic programming	8	CO2
3.	**Graph algorithms:** Traversal, topological sort, minimum spanning trees, shortest path,biconnected components, strongly connected components in directed graphs, network flow	8	CO3
4.	**NP-completeness:** Classes P, NP and space complexity, reduction, NP-completeness, examples of NP-complete problems	6	CO4
5.	**Geometric algorithms:** Convex hulls, sweep paradigm, Voronoi diagrams, closest pair, nearest neighbour search.	6	CO5
6.	**Approximation Algorithms:** Approximation algorithms for known NP complete problems	6	CO6

Text Books	
1.	Kleinberg, Jon, and Eva Tardos. Algorithm design. Pearson Education India, 2006.
2.	Cormen, Thomas H., Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. Introduction to algorithms. MIT press, 2022.
Reference Books	
1.	Aho, A., J. Hopcroft, and J. Ullman. ”The Design and Analysis of Algorithms. Addison and Wesley.” Reading, MA (1974).

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5012S
Course Title	Cloud Computing
Course Type	Program Core
Prerequisites: Operating system	
Course Outcomes: At the end of the course student will be able to: **CO1.** Describe cloud computing concepts, architecture, deployment model and other cloud terminologies **CO2.** Use various aspects of SLA, workout cloud economics. **CO3.** Apply security features of cloud. **CO4.** Distinguish various special purpose cloud. **CO5.** Use and Create private cloud. **CO6.** Design solution using cloud services.	

	Course Contents	Hrs.	CO
1.	**Introduction:** Cloud Computing – Overview, Cloud Computing – Introduction, Evolution of Cloud Computing, Cloud Computing Architecture, Cloud Architecture - Deployment Models, , Virtualization, XML Basics, Web Services, Service Oriented Architecture	6	CO1
2.	**Service Level Agreement (SLA):** SLA – Tutorial (problems), Economics, Economics Tutorial (problems), Managing Data, Introduction to Map Reduce, Map Reduce – Tutorial (problem), Resource Management in cloud	8	CO2
3.	**Openstack Cloud:** Detail Study of openstack cloud, Deployment/ implementation of Openstack cloud as private cloud, OpenStack Major Components, Architecture of Openstack, Openstack Work Flow, Nova scheduler filtering, OpenStack Storage Concepts Study of other services	8	CO5
4.	**Cloud Types:** Broker for Cloud Marketplace, Fog Computing, Use Case Geospatial Cloud, Green Cloud, Sensor Cloud Computing, IoT Cloud	4	CO4

5.	**Cloud Security:** Security - Basic Components, Security Attacks, Classes of Threats, Policies and Mechanisms, Goals of Security, Trust and Assumptions, Types of Mechanisms, Assurance, Operational Issues, Passive and Active Attacks, Security Services.	8	CO3
6.	**Cloud Platforms study:** Case study of public cloud like AWS, Microsoft Azure, Google cloud GCP, Study and use of various services. Cloud Migration, Docker container, Serverless Computing, Dew computing	6	CO6

Text Books	
1.	Arshdeep Bahga , Vijay Madisetti, Cloud Computing: A Hands-On Approach. Universities Press.
Reference Books	
1.	Buyya, Rajkumar, Christian Vecchiola, and S. Thamarai Selvi. Mastering cloud computing: foundations and applications programming. Newnes, 2013.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5061S
Course Title	Database Management Systems
Course Type	Open Elective
Prerequisites: Nil.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Differentiate various database architectures. **CO2.** Design and implement relational databases. **CO3.** Execute MongoDB commands to manipulate data. **CO4.** Use advanced XML queries on database. **CO5.** Apply practices of implementing database security. **CO6.** Perform transaction processing and achieve concurrency control.	

	Course Contents	Hrs.	CO
1.	**Introduction to Database Management Systems and Enhanced Data Models for Advanced Applications:** Characteristics of database, Database users, Advantages of DBMS, Data Models, Schema and Instances, Three schema Architecture and Data Independence, Database Languages and Interfaces, The Database System Environment, Centralized and Client / Server Architecture for DBMS. Introduction to Temporal Database and Multimedia Databases.	4	CO1
2.	**Relational-Database Design and SQL:** Functional dependencies, Normalisation forms, Decomposition, Overall database design process. SQL: DDL: Create, Modify, Alter, Drop, View definition, etc.DML: SELECT, INSERT, DELETE, Update, Nested Query, SQL with SET operations: Union, Intersect, Except, etc, Aggregate Functions: Group By, Having, SUM, etc, SQL with Logical operations, Nested and Complex Queries, Join Queries. DCL : GRANT, REVOKE, etc DBA level query. Cursors and Triggers, Procedures and Functions, Partitions, SQL Backup and Recovery.	8	CO2

3.	**Introduction to NoSQL Databases:** Introduction, Design of parallel systems, Parallel query processing. Avenues for parallelism, Array and vector processors. Multiprocessor architecture: taxonomy of parallel architectures, Parallel Query Evaluation. Advanced Transaction Processing Non-relational DBMS: consistency and availability trade-offs, NoSQL DBMS (key-value, document, and graph), MongoDB: CRUD operations.	8	CO3
4.	**XML Databases:** Introduction to XML Documents and Databases, XML schemas, tree structure, and DOM, XML Query.	6	CO4
5.	**Database Security:** Introduction to major database attacks: SQL Injection, DoS/DDoS etc. Encryption and Public Key Infrastructures.	4	CO5
6.	**Transaction Processing and Concurrency Control:** Schedules and serializability, Lock management, Compensation and Databases, Deadlock Handling, Multiple granularity, validation protocols, multi-version protocols, snap shot isolation, predicate locking, Weak Levels of Consistency in Practice.	8	CO6

Text Books	
1.	Elmasri, Navathe. Fundamentals of Database Management systems, Pearson Education, 2008.
2.	Avi Silberschatz, Henry F. Korth, S. Sudarshan. Database System Concepts, Seventh Edition, McGraw-Hill, 2010.
3.	P. Sadalage and M. Fowler, NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Addison Wesley, 2012.
Reference Books	
1.	Thomas Connolly and Carolyn Begg, "Database Systems" 3rd Edition, Addison-Wesley, 2005.
2.	V.S. Subrahmanian, "Multimedia database systems", Springer, 1996.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5063S
Course Title	Data Science
Course Type	Open Elective
Prerequisites: Nil.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Describe a flow process for data science problems. **CO2.** Classify data science problems into standard typology **CO3.** Develop R codes for data science solutions **CO4.** Correlate results to the solution approach followed. **CO5.** Assess the solution approach **CO6.** Construct use cases to validate approach and identify modifications required.	

	Course Contents	Hrs.	CO
1.	**Foundations Linear algebra for data science: Algebraic view** - vectors, matrices, product of matrix and vector, rank, null space, solution of over- determined set of equations and pseudo-inverse) **Geometric view** - vectors, distance, projections, eigenvalue decomposition.	4	CO1
2.	**Statistics:** descriptive statistics, notion of probability, distributions, mean, variance, covariance, covariance matrix, understanding univariate and multivariate normal distributions, introduction to hypothesis testing, confidence interval for estimates	8	CO2
3.	**Optimization Techniques:** Optimization, Typology of data science problems and a solution framework	8	CO3
4.	**Regressions:** Simple linear regression and verifying assumptions used in linear regression, Multivariate linear regression, model assessment, assessing importance of different variables, subset selection.	6	CO4
5.	**Classification:** Classification using logistic regression, kNN and k-means clustering	4	CO5, CO6

Text Books	
1.	Ragunathan Rengasamy and Shankar Narasimhan , Data Science for Engineering , Computer Science and Engineering, IIT Madras.
2.	Reema Thareja ,Data Science and Machine Learning using Python, McGrawHill Publication, 2021
Reference Books	
1.	Avi Silberschatz, Henry F. Korth, S. Sudarshan. Database System Concepts, Seventh Edition, McGraw-Hill, 2010.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5062S
Course Title	Data Structure
Course Type	Open Elective
Prerequisites: Nil.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Apply basic data structures such as various linear and nonlinear data structures, CO2. CO3. CO4. **CO2.** Analyse use of basic data structures in different applications and implement various applications and solve exercises using appropriate data structures. **CO3.** Evaluate methods for analysis of algorithms, learn, analyze and implement different searching and sorting techniques and their implementation **CO4.** Understand and apply fundamental algorithmic problems including Tree traversals, Graph theory and hashing techniques	

	Course Contents	Hrs.	CO
1.	**Introduction to Data structures and Analysis of Algorithms:** Need of data structures, Types of data structures, recursion, ADT (Abstract Data Types). Basics of algorithm, analysis of algorithm through time complexity and space complexity, asymptotic notations, pseudo code analysis, various examples to show time complexity calculation. Recurrence Relations and Solving Recurrences Using Substitution, Recursion Tree and Master Method	8	CO1
2.	**Stack and Queue:** Stack: The stack as an ADT, Representation, Stack operation, Application. Queue: The Queue as an ADT, Representation, Queue operation, Circular and Priority queue, Applications.	8	CO2
3.	**Linked List:** Linked list as an ADT, Operation on linked list, Linked stacks and Queues, Array implementation of Linked List, Linked list using Dynamic Variable, Doubly, circular linked list.	8	CO3

4.	**Binary Tree:** Basic tree concept, Binary tree operations, Binary tree representation, Binary tree traversals , Binary search tree and operations on it, balanced tree: AVL trees and operations, applications of these binary trees and exercises on it. Implementing priority queue using binary heap data structure.	6	CO4
5.	**Graphs:** Basics concepts of graphs, representation of graphs, graph traversals BFS and DFS, minimum spanning tree algorithms: Kruskal's algorithm and Prim's algorithm, application and related exercises in brief.	4	CO5
5.	**Searching Techniques and Hashing:** Linear Search and Binary Search, Hashing: Direct-address tables, Hash tables, open addressing, Perfect Hashing, Sorting techniques: Various sorting methods and their time complexity analysis: Insertion sort, selection sort ,merge sort, quick sort, heap sort,	4	CO6

Text Books	
1.	Y. Langsam, M. J.Augenstein and A. M. Tanenbaum "Data structures using Java", Second edition 2007 Pearson Education
2.	T. H. Cormen, C. E. Leiserson, R. L.Rivest and C. Stein, "Introduction to Algorithms", Third Edition, MIT Press/McGraw Hill.
3.	Goodrich and Tamassia, Data Structures and Algorithm in Java, John Wiley and Sons, Sixth Edition 2014
Reference Books	
1.	Avi Silberschatz, Henry F. Korth, S. Sudarshan. Database System Concepts, Seventh Edition, McGraw-Hill, 2010.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5071L
Course Title	Laboratory-1: Cloud Computing
Course Type	Laboratory
Prerequisites: Nil.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Create Virtual machines and use IaaS services of cloud like AWS/Azure/GCP **CO2.** Design and deploy a web application using Pass facility **CO3.** Create and use Container instances, Virtual Network, Object Storage, Databases	

It is expected to conduct 8-10 experiments in the field of knowledge. One hour shall be tutored about theory and relevant tools to the students, and students shall perform the experiment. This is continuous evaluation, hence all experiments shall be evaluated in the same week. A sample list of experiments is given it may be altered as and when required.

Sr. No.	Course Contents
1.	Create a Virtual Machine in the Portal (AWS/Azure/Google)
2.	Create a Web App (AWS/Azure/Google)
3.	Deploy Container Instances (AWS/Azure/Google)
4.	Create a Virtual Network (AWS/Azure/Google)
5.	Create Object Storage (AWS/Azure/Google)
6.	Create a Database (AWS/Azure/Google)
7.	Implement an Azure IoT Hub /AWS IoT Core/Google IoT
8.	Implement Azure Functions /AWS Lamda /Google serverless
9.	Create a VM with a Template
10.	Create a VM with PowerShell / Command Line

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5072L
Course Title	Laboratory-2: Software Testing
Course Type	Laboratory
Prerequisites: Software Engineering.	
Course Outcomes: At the end of the course student will be able to: **CO1.** To provide knowledge of Version control system using GitHub. **CO2.** To provide knowledge of Software Testing Methods. **CO3.** To develop skills in software test automation and management using latest tools.	

It is expected to conduct 8-10 experiments in the field of knowledge. One hour shall be tutored about theory and relevant tools to the students, and students shall perform the experiment. This is continuous evaluation, hence all experiments shall be evaluated in the same week. A sample list of experiments is given it may be altered as and when required.	
Sr. No.	Course Contents
1.	Create GitHub Repository, Clone sample project and Create new branch
2.	Update given code and committee your code changes
3.	Prepare software test plan for the software
4.	Write Unite test cases and carryout functional testing of the software
5.	Execute test cases for Structural testing of the software and prepare code coverage report
6.	Generate automated test cases using tools like selenium
7.	Generate regression test case suite for the software
8.	Generate regression test test report for the software
9.	Generate stress test test report using JMeter for the software
10.	Generate Quality Assurance Plan for the software

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5073L
Course Title	Laboratory-3: Cross Platform App Development
Course Type	Laboratory
Prerequisites: Software Engineering, Web Development.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Analyse the solutions for cross-platform design and development challenges **CO2.** Analyse components for user interface design **CO3.** Implement data and program flow for complex application. **CO4.** Integrate Rest APIs and deploy on various platforms.	

It is expected to conduct 8-10 experiments in the field of knowledge. One hour shall be tutored about theory and relevant tools to the students, and students shall perform the experiment. This is continuous evaluation, hence all experiments shall be evaluated in the same week. A sample list of experiments is given it may be altered as and when required.	
Sr. No.	Course Contents
1.	Designing user interface and wireframe for multiple layouts using Figma.
2.	Setting up the development environment of React Native and .NET MAUI on Windows and building first simple application.
3.	Programs regarding concepts of JavaScript, JSX, XAML and C#.
4.	Basic user interface design using react components, Stylesheet, Flexbox and XAML, manipulation of components using States and Props.
5.	Adding user interactivity in the base components and customization of layout using functions provided by React Native and XAML.
6.	Integrating third party components into our application and XAML pages using shared resources.
7.	Implementing multiple screen navigation and nested navigation using solutions provided by React Navigation and .NET MAUI.

8.	Implementing data communication between multiple screens using props and call-back functions.
9.	Implementation of asynchronous functions and outside API calls in JavaScript, react Native and .NET MAUI.
10.	Exploration of life cycle functions for components, hook functions and XAML pages.
11.	Implementation and integration with firebase backend and application deployment.

Programme Name	M. Tech. Computer Engineering (Specialization in Network Infrastructure and Management Systems)
Course Code	COCE5021T
Course Title	Computer Network Design
Course Type	Program Elective
Prerequisites: Nil.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Apply networking tools and techniques to design computer network. **CO2.** Evaluate Transport Layer protocols for Quality of Service (QoS). **CO3.** Examine Network Layer services and Protocols. **CO4.** Recommend Application Protocols as per need of application. **CO5.** Categorize various network security flaws. **CO6.** Design and deploy computer network as per customer requirement.	

	Course Contents	Hrs.	CO
1.	**Computer Network Design Foundation:** Introduction to Networking: Comparison between OSI and TCP/IP Protocol Suite, IP addressing, Cables, Repeaters, Bridges, Routers, Switches, Hubs, Gateway, VLANS, Network Design Case studies. **Designing Secure Networks**: Device hardening network security platform options and best deployment practices, common application design considerations, Identity design consideration, IPsec VPN design consideration, Network Design Case studies.	6	CO1, CO6
2.	**Transport Layer:** Introduction, Transport Layer Protocols, Congestion Control and Quality of Service User Datagram Protocol (UDP), Transmission Control Protocol (TCP), Stream Control Transmission Protocol (SCTP)	10	CO2, CO6
3.	**Network Layer:** Introduction, packet format, IPV4 addresses, Internet protocol Version 4(IPV4), IPV6, Address resolution protocol (ARP), Reverse address resolution protocol (RARP), Internet control Message protocol (ICMP), Real time transport protocol (RTP), RTP control protocol (RTCP), VOICE OVER IP etc	7	CO3, CO6

4.	**Routing Protocols:** Unicast Routing Protocols (RIP, OSPF, and BGP), Multicasting and Multicast Routing Protocols, RIP (Routing information protocol), OSPF (Open shortest path first), BGP (Border gateway protocol), Internet group management protocol (IGMP)	7	CO3, CO6
5.	**Application Layer Protocols:** Introduction, Host Configuration: BOOTP and DHCP, Domain Name System (DNS), TELNET and SSH, File Transfer: FTP and TFTP, SNMP	5	CO4, CO6
6.	**Applications of Secure Computer Network:** Security policy and operation life cycle, Internet Security: Security aspects in Network Layer, Transport Layer, and Application Layer, working of Firewalls,	4	CO5
	Advances in the domain	4	CO5

Text Books	
1.	Behrouz A. Forouzan, "TCP/IP Protocol Suite", III Edition, Tata McGraw Hill, 2005
2.	Behrouz A. Forouzan, "Data Communications and Networking", Tata McGraw-Hill, Fourth Edition
Reference Books	
1.	W. Richard Stevens, "TCP/IP Illustrated, Volume 1", Addison-Wesley, Second Edition
2.	Internetworking with TCP-IP: Design, Implementation, and Internals, by D. E. Comer and D. L. Stevens Vol II, Prentice Hall.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5022T
Course Title	Distributed Systems
Course Type	Program Elective
Prerequisites: Operating Systems, Computer Networks.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Illustrate fundamental concepts of distributed systems. **CO2.** Demonstrate synchronization principles for real world problems in distributed systems. **CO3.** Distinguish different middle-ware technologies in computer distributed applications. **CO4.** Examine shared data operations and replication in the distributed applications. **CO5.** Inspect distributed systems with case studies. **CO6.** Design advanced graph theoretic algorithms.	

	Course Contents	Hrs.	CO
1.	**Foundations:** Examples of distributed systems, Architectural models, Network principles, Multi-cast communication, Network virtualization, Message passing interface (MPI), Request-reply protocols, Remote procedure call (RPC), Remote method invocation (RMI), group communication, publish-subscribe systems, message queue systems, shared memory–based approaches.	9	CO1
2.	**Synchronization and Coordination of Distributed Systems:** Clocks, events and process states, Synchronizing physical clocks, Logical time and logical clocks, Global states, Distributed mutual exclusion, Elections, Coordination and agreement in group communication, Consensus and related problems	8	CO2
3.	**Middle-ware Components:** Distributed objects, CORBA, Distributed components, Enterprise JavaBeans and Fractal, Web services, Coordination of web services, Peer-to-peer middleware, Routing overlays, Overlay case studies: Pastry, Tapestry	6	CO3

4.	**Distributed Shared Data:** Distributed mutual exclusion, Elections, Coordination and agreement in group communication, Consensus, Transactions, Nested transactions, Locks, Optimistic concurrency control, Timestamp ordering, Flat and nested distributed transactions, Atomic commit protocols, Concurrency control in distributed transactions, Distributed deadlocks, Transaction recovery, System model and the role of group communication, Fault-tolerant services, Case studies of highly available services: The gossip architecture, Bayou and Coda, Transactions with replicated data	9	CO4
5.	**Distributed Systems:** Overview of security techniques, Case studies: Needham–Schroeder, Kerberos, TLS, 802.11 WiFi, File service architecture, Case study: Sun Network File System, Case study: The Andrew File System, Name services and the Domain Name System, Directory services, Case study: The Global Name Service, Case study: The X.500 Directory Service	6	CO5
6.	**Designing Distribute Systems:** Introducing the case study: Google, Overall architecture and design philosophy, Underlying communication paradigms, Data storage and coordination services, Distributed computation services	3	CO6

Text Books	
1.	George Coulouris, Jean Dollimore, and Tim Kindberg, “ Distributed Systems Concepts and Design”, 5th ed., Pearson Education, 2011.
2.	Ghosh, Sukumar. Distributed systems: an algorithmic approach. Chapman and Hall/CRC; 2nd edition 2014.
Reference Books	
1.	Van Steen, Maarten, and Andrew S. Tanenbaum. Distributed systems. Leiden, The Netherlands: Maarten van Steen, 2017.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5023T
Course Title	Computer Systems Performance Analysis
Course Type	Program Elective
Prerequisites: Operating Systems, Computer Networks, and Probability and Statistics.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Evaluate the performance of the various computer systems and networks, mistakes commonly observed in performance evaluation projects and a proper methodology to avoid them. **CO2.** Illustrate performance metrics and analyze them mathematically, analytically or through simulation. **CO3.** Illustrate the application of probability functions and distributions for computer systems. **CO4.** Design mathematical models using Queuing Networks models and use programming languages to simulate and evaluate the performance of various computer systems within or outside the network. **CO5.** Evaluate the performance of network systems by using various queuing models. **CO6.** Analyze the performance evaluation of various protocols, algorithm in an network environment.	

	Course Contents	Hrs.	CO
1.	**Overview of Performance Evaluation:** Introduction, common mistakes and how to avoid them, selection of techniques and metrics.	10	CO1
2.	**Measurement Techniques and Tools:** Types of workloads, the art of workload selection, workload characterization and techniques, monitors, program-execution monitors and accounting logs, capacity planning and bench-marking, the art of data presentation.	10	CO2
3.	**Probability Theory and use for Evaluation:** Introduction to probability refresher, conditional probability, total probability, discrete and continuous random variables, common distributions, probability generating functions(pgf) and Laplace transforms (lst), numerous examples from computer networking, Commonly used distributions.	6	CO3

4.	**Queuing Theory :** Queuing models, little theorem application, stochastic processes, Markov chain formulation, discrete time and continuous time markov chains (dtmc, ctmc), MMD, Operational laws.	4	CO4
5.	**Queuing System Models and Application:** Queuing system m/m/1, m/m/1/k, m/m/s/, m/m/y queue analysis m-server case. Multidimensional markov chain application in circuit switching/g/1 queue, generalization of m/g/1 theory application to atm, embedding instants in the m/g/1 theory m/g/1 with geometrically distributed messages. chain embedded to cell transmission, message transmission completion. queue balance equation, finite buffer case, mean value analysis.	6	CO5
6.	**Network Analysis:** Local area Network analysis, standard comment based analysis, contention based protocols, demand assignment protocols, nodes in packet switches networks, performance analysis of data link layer, Network layer. Traffic control and congestion in ATM networks, TCP/IP Traffic control.	3	CO6

Text Books	
1.	Raj Jain, "The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling", Wiley-Interscience, 1991.
Reference Books	
1.	K.S. Trivedi, "Probability and Statistics with Reliability, Queueing and Computer Science Applications", John Wiley and Sons, 2001.
2.	Ross, Sheldon M. Introduction to probability models. Academic press, 2014.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5024T
Course Title	Social Network Analysis
Course Type	Program Elective
Prerequisites: Analysis of Algorithm, Computer Networks, Data Mining.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Illustrate sociology and anthropology used the ideas of culture and cultural formation concepts of Social Network Analysis. **CO2.** Demonstrate statistical properties for real world problems in Social Network Analysis. **CO3.** Distinguish different edge classification techniques and use in the application. **CO4.** Examine shared data operations and replication in the distributed applications. **CO5.** Inspect data mining in Social Network Analysis with case studies. **CO6.** Design advanced graph theoretic algorithms.	

	Course Contents	Hrs.	CO
1.	**Foundations:** Introduction to new science of networks; Networks examples; Sociometry, small groups, and communities; Cliques, roles, and matrices; Space and distance; Dynamics and social change	6	CO1
2.	**Statistical Properties of Social Network:** Graph concepts and properties; static and dynamic properties of social graphs; RandomWalk based Proximity Measures; other proximity measures; Graph-theoretic Measures for Semi-supervised Learning; Algorithms for computing the proximity measures; Applications using random walks approach.	8	CO2
3.	**Community Discovery in Social Networks:** Defining communities; Core Methods: Quality Functions, Kernighan-Lin(KL) algorithm, Agglomerative/Divisive Algorithms, Spectral Algorithms, Multi-level Graph Partitioning, Markov Clustering; Community Discovery in Dynamic Networks, Heterogeneous Networks, Directed Networks.	6	CO3

4.	**Node Classification in Social Networks:** The Node Classification Problem; Problem Formulation; Local Classifiers, Random Walk based classifier, Node Classification to Large Social Networks, Basic Methods, Second-order Methods, Map-Reduce, Dissimilarity in Labels, Edge Labeling, Label Summarization.	6	CO4
5.	**Data Mining in Social Media:** Data Representation, Event Maps, Social Networking Sites: Illustrative Examples, Blogosphere: Illustrative Examples.	6	CO5
6.	**Visualizing Social Network:** Visual Images, MDS and SVD to explore data, Exploratory Research, Validating a Model, Structural Visualization, Semantic and Temporal Visualization, Statistical Visualization	6	CO6

Text Books	
1.	Charu C. Aggarwal, ”Social Network Data Analytics” Springer New York, NY, 2011.
2.	Carrington, P., Scott, J., Wasserman, S. (Eds.). Models and Methods in Social Network Analysis (Structural Analysis in the Social Sciences). Cambridge: Cambridge University Press, 2005.
Reference Books	
1.	Xiaoming Fu, Jar-Der Luo, Margarete Boos. Social Network Analysis Interdisciplinary Approaches and Case Studies.CRC Press, 2020.
2.	Carrington, Peter J., John Scott, and Stanley Wasserman, eds. Models and methods in social network analysis. Vol. 28. Cambridge university press, 2005.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5031S
Course Title	Software Engineering
Course Type	Program Elective
Prerequisites: Programming, basics of software engineering	
Course Outcomes: At the end of the course student will be able to: **CO1.** Illustrate software development process and best practice for software development. **CO2.** Demonstrate agile development principles and techniques to manage agile software developments. **CO3.** Investigate requirements to generate software requirement document. **CO4.** Apply component based design to real life applications. **CO5.** Illustrate skills of software testing to projects. **CO6.** Estimate cost of the project.	

	Course Contents	Hrs.	CO
1.	**Fundamentals of Software Engineering:** software life-cycle process models, industry-standard software engineering tools.	3	CO1
2.	**Agile Methods for Software Development Method:** Extreme Programming (XP), Scrum, Lean, Crystal, Dynamic Systems Development Method and Feature-Driven Development.	8	CO2
3.	**Software Requirements Analysis and Engineering:** Identification of stakeholders, the elicitation and verification of requirements from them, and translation into detailed requirements	6	CO3
4.	**Software Architecture and Component-Based Design:** software design process and it's models; software architectures and design plans; design methods; design state assessment; design quality assurance; and design verification.	8	CO4
5.	**Software Testing and Quality Assurance:** systematic testing of software systems, software verification, symbolic execution, software debugging, quality assurance, measurement and prediction of software reliability.	8	CO5

6.	**Cost Estimation and Measurement:** industry-standard software sizing metrics as Function, Feature, and Object Points and their relationship to the lines-of-code metric.	6	CO6

Text Books	
1.	Roger Pressman. Software Engineering: A Practitioner's Approach (7th. ed.). McGraw-Hill, Inc., USA. 2009.
Reference Books	
1.	Jalote, Pankaj. An integrated approach to software engineering. Springer Science & Business Media, 2012.
2.	Sommerville, Ian. Software Engineering, 9/E. Pearson Education India, 2011.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5032S
Course Title	Internet of Things
Course Type	Program Elective
Prerequisites:	
Course Outcomes: At the end of the course student will be able to: **CO1.** Analyze the importance of various aspects of IoT. **CO2.** Apply different tools for interoperability for IoT. **CO3.** Design different SDN for IoT. **CO4.** Analyze Data Handling and Analytics in IoT. **CO5.** Design and develop different application in IoT. **CO6.** Create different case studies in IoT.	

	Course Contents	Hrs.	CO
1.	**Fundamentals of Internet of Things:** Introduction to IoT, Sensing, Actuation, Basics of Networking, Communication Protocols, Sensor Networks, Machine-to-Machine Communications.	4	CO1
2.	**Interoperability in IoT:** Introduction to Arduino Programming, Integration of Sensors and Actuators with Arduino, Introduction to Python programming, Introduction to Raspberry Pi.	6	CO2
3.	**Implementation of IoT:** Implementation of IoT with Raspberry Pi, Introduction to SDN, SDN for IoT.	8	CO3
4.	**Data Handling in IOT:** Data Handling and Analytics, Cloud Computing, Cloud Computing, Sensor-Cloud, Fog Computing.	8	CO4
5.	**Application of IOT:** Connected Vehicles, Smart Grid, Industrial IoT. Challenges in Design and Development		CO5
6.	**Case Study:** Agriculture, Healthcare, Activity Monitoring.		CO6

Text Books	
1.	S. Misra, A. Mukherjee, and A. Roy, 2020. Introduction to IoT. Cambridge University Press.
2.	S. Misra, C. Roy, and A. Mukherjee, 2020. Introduction to Industrial Internet of Things and Industry 4.0. CRC Press.
Reference Books	
1.	Research Papers

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5033S
Course Title	GPU Architecture and Programming
Course Type	Program Elective
Prerequisites: Nil.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Describe concepts of the GPU architecture. **CO2.** Justify memory hierarchy and usage in parallel programming. **CO3.** Apply synchronization primitives in parallel programming. **CO4.** Justify data transfer through streams for parallel programs. **CO5.** Infer kernel functions for the real time systems. **CO6.** Illustrate case studies.	

	Course Contents	Hrs.	CO
1.	**Introduction:** Streaming Multi Processors, Cache Hierarchy,The Graphics Pipeline History, Graphics processors, graphics processing units, Clock speeds, CPU/GPU comparisons, heterogeneity, Accelerators, Parallel programming, CUDA / OpenCL / OpenACC	8	CO1
2.	**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.	6	CO2
3.	**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. Warp Scheduling, Divergence	8	CO3
4.	**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	CO4

5.	**Functions:** Device functions, host functions, kernels, functors, Optimization examples : optimizing Reduction Kernels Optimization examples : Kernel Fusion, Thread and Block OpenCL basics and OpenCL for Heterogeneous Computing Support: Debugging GPU programs. Profiling, profile tools, performance aspects	8	CO5
6.	**Advanced topics:** Case studies, Dynamic Parallelism, Unified virtual memory, Multi-GPU processing, Peer access, Heterogeneous processing	6	CO6

Text Books	
1.	Programming Massively Parallel Processors: A Hands-on Approach; David Kirk, Wen-mei Hwu; Morgan Kaufman; 2016
Reference Books	
1.	CUDA Programming: A Developer's Guide to Parallel Computing with GPUs; Shane Cook; Morgan Kaufman; 2012
2.	Heterogeneous Computing with OpenCL" – Benedict Gaster,Lee Howes, David R. Kaeli

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5034S
Course Title	Graph Mining
Course Type	Program Elective
Prerequisites: Fundamentals of probability and linear algebra.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Demonstrate deep knowledge of the fundamentals of graph matching, visualization and data mining. **CO2.** Design and analyze the graph patterns for graph based application. **CO3.** Finding topological frequent patterns from graph database. **CO4.** Develop efficient and salable implementations of graph mining algorithms. **CO5.** Design a decision tree based on chunkingless graph based induction for graph application. **CO6.** Applying graph mining knowledge to design effective solution to real-life graph analytic problems	

	Course Contents	Hrs.	CO
1.	**Fundamentals of Graph Mining:** Terminology, Graph Databases Graph Matching—Exact And Error-Tolerant Methods And The Automatic Learning Of Edit Costs : Introduction, Definitions and Graph Matching Methods, Learning Edit Costs, Experimental Evaluation. Graph Drawing Techniques, Examples of Visualization Systems	4	CO1
2.	**Graph Patterns And The R-Mat Generator:** NetMine and R-MAT, Experiments Discovery Of Frequent Substructures : Introduction, Preliminary Concepts, Apriori-based Approach, Pattern Growth Approach, Variant Substructure Patterns, Experiments and Performance Study DFS Approach (gSpan and others), Diagonal and Greedy Approaches, Constraint-based mining and new algorithms, Mining Frequent Subgraphs.	8	CO2

3.	**Finding Topological Frequent Patterns From Graph Datasets:** Frequent Pattern Discovery from Graph Datasets, Problem Definitions, FSG for the Graph-Transaction Setting, SIGRAM for the Single-Graph Setting, GREW, Scalable Frequent Subgraph Discovery Algorithm.	6	CO3
4.	**Unsupervised And Supervised Pattern Learning In Graph Data:** Mining Graph Data Using Subdue, Comparison to Other Graph-Based Mining Algorithms, Comparison to Frequent Substructure Mining Approaches, Comparison to ILP Approaches Graph Grammar Learning : Introduction, Related Work, Graph Grammar Learning, Empirical Evaluation	6	CO4
5.	**Constructing Decision Tree Based On Chunkingless Graph-Based Induction:** Graph-Based Induction Revisited, Problem Caused by Chunking in B-GBI, Chunkingless Graph-Based Induction (Cl-GBI), Decision Tree Chunkingless Graph-Based Links Between Formal Concept Analysis And Graph Mining : Presentation, Basic Concepts and Notation, Formal Concept Analysis, Extension Lattice and Description Lattice Give Concept Lattice, Graph Description and Galois Lattice, Graph Mining and Formal Propositionalization, Kernel Methods For Graphs : Introduction, Graph Classification, Vertex Classification.	8	CO5
6.	**Applications of Graph Mining:** Web mining, centrality analysis, Link analysis algorithms, graph clustering and community detection, Node classification and Link prediction, Influential spreaders, Influence maximization, Geo-social and location based networks.	6	CO6

Text Books	
1.	Diane J. Cook , Lawrence B. Holder, Mining Graph Data, Wiley Publication, 2010.
Reference Books	
1.	Deepayan Chakrabarti and Christos Faloutsos, Graph Mining: Laws, Tools, and Case Studies, Synthesis Lectures on Data Mining and Knowledge Discovery, Morgan & Claypool Publishers, 2012
2.	Charu C. Agrawal, Haixun Wang, Managing and Mining Graph Data, Springer, 2012.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5002S
Course Title	Research Methodology and IPR
Course Type	Program Core (Mandatory Learning)
Prerequisites: Nil.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Explore research and describe the research process and research methods. **CO2.** Model and visualize the processes and requirements for conducting successful research. **CO3.** Identify the requirement of report writing and apply over it. **CO4.** Investigate and apply the basic aspects of the scientific conduct and publication ethics in order to demonstrate through software approach. **CO5.** Apply knowledge in publication ethics and investigate misconduct for the exploration of required IT Acts in research project. **CO6.** Able to present, review and publish on scientific paper work.	

	Course Contents	Hrs.	CO
1.	**Research Methods:** Objectives of Research, Various Steps in Scientific Research, Types of Research; Research Problem , Research Design , Survey Research , Case Study Research and hypothesis,Sampling , Measurement and Scaling techniques , Methods of data collection, Design of Survey and Experiments , Hypothesis design	6	CO1
2.	**Computer Application in Research Methodology:** C Data Processing and Modeling :Data processing and Measures Mathematical model formulation for queries using relational algebra, Design of software Architecture ,Database design, Algorithm Design ,GUI design ,Model building and decision making , Probability Distributions, Fundamentals of Statistical Analysis and Inference, Correlation and Regression ,Classification ,Clustering	6	CO2

3.	**Report writing:** Structure and Components of Research Report, Types of Report, Layout of Research Report, Writing research proposal ,Mechanism of writing a research report, Performance evaluation and curve fitting, Result declaration by various graphs and charts	6	CO3
4.	**Scientific conduct and publication ethics:** Ethics with respect to science and research,Intellectual honesty and research integrity, falsification, fabrication and plagiarism., duplicate and overlapping publication, salami slicing,Selective reporting and misinterpretation of data,Best practices/standards setting initiatives and guidelines: COPE, WAME, etc,Publication misconducts: definition, concept, problems that lead to unethical behavior and vice versa, types,Violation of publications ethic, authorship and contribution ship, Identification of publication misconduct, complaints and appeals,Software tools: Use of plagiarism tool like Turnitin, Urkund, and other open-source software tool	6	CO4
5.	**Publication ethics and misconduct:** Subject specific ethical issues, FFP, authorship,Conflict of interest,Complaints and appeals: examples and fraud from India and abroad,IT Acts for handling misconduct.	6	CO5
6.	**Application Of Results and Ethics:** Ethical issues , ethical committees , Commercialization , Code of Research Ethics Intellectual property Trademark rights ,Copyright,Patent ,Plagiarism ,royalty Databases Indexing databases, Citation databases: Web of science, Scopus etc. Research Metrics:Impact factor of journal as per citation report, SNIP, SJR, IPP, Cite score ,Metrics: h-index, i10-index, g-index, altmetrics	8	CO6

Text Books	
1.	G.C.Ramamurthy, “Research Methodology”, Dreamtech Publications, 2015.
2.	C. R. Kothari, “Research Methodology: Methods and Techniques” fourth Edition, New Age International Publishers, New Delhi, 2019.
Reference Books	
1.	A. Ravindran, K. M. Ragsdell, G. V. Reklaitis, “Engineering Optimization methods and applications”, Second Edition, Wiley publication, 2006.
2.	Donald R. Cooper, Pamela S. Schindler, “Business Research Methods”, eighth edition, Tata McGraw-Hill, 2006.
3.	MacIntyre, A., “A Short History of Ethics: a history of moral philosophy from the Homeric age to the 20th century”, Second Edition, 2003.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5013T
Course Title	Big Data Analytics
Course Type	Program Core-III
Prerequisites: Database Management System, Java, Python, AI, Machine Learning.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Analyse the important components of big data. **CO2.** Illustrate different components in Hadoop and MapReduce. **CO3.** Develop problem solving skills like Collect, manage, store, query, and analyze various forms of Big Data using NoSQL. **CO4.** Solve different problems using data streaming in big data analytics. **CO5.** Apply adequate perspectives of big data analytics in various applications like recommender systems and social media applications. **CO6.** Analyse spark framework of big data analytic.	

	Course Contents	Hrs.	CO
1.	**Introduction to Big Data and Hadoop:** Introduction to Big Data, Big Data characteristics, types of Big Data, Traditional vs. Big Data business approach, Case Study of Big Data Solutions,Concept of Hadoop Core Hadoop Components; Hadoop Ecosystem	4	CO1
2.	**Hadoop HDFS and MapReduce:** Distributed File Systems: Physical Organization of Compute Nodes, Large-Scale File-System Organization. MapReduce: The Map Tasks, Grouping by Key, The Reduce Tasks, Combiners, Details of MapReduce Execution, Coping with Node Failures. Algorithms Using MapReduce: Matrix-Vector Multiplication by MapReduce, Relational-Algebra Operations, Computing Selections by MapReduce, Computing Projections by MapReduce, Union, Intersection, and Difference by MapReduce Hadoop Limitations	4	CO2

3.	**NoSQL:** Introduction to NoSQL, NoSQL Business Drivers, NoSQL Data Architecture Patterns: Key-value stores, Graph stores, Column family (Bigtable)stores, Document stores, Variations of NoSQL architectural patterns, NoSQL Case Study NoSQL solution for big data, Understanding the types of big data problems; Analyzing big data with a shared-nothing architecture Choosing distribution models: master-slave versus peer-to-peer; NoSQL systems to handle big data problems	6	CO3
4.	**Mining Data Streams:** The Stream Data Model: A Data-Stream-Management System, Examples of Stream Sources, Stream Queries, Issues in Stream Processing. Sampling Data techniques in a Stream Filtering Streams: Bloom Filter with Analysis,Counting Distinct Elements in a Stream, Count-Distinct problem, Flajolet-Martin Algorithm, Combining Estimates, Space Requirements Counting Frequent Items in a Stream, Sampling Methods for Streams, Frequent Item sets in Decaying Window,Counting Ones in a Window: The Cost of Exact Counts, The Datar-Gionis-Indyk-Motwani Algorithm, Query Answering in the DGIM Algorithm, Decaying Windows.	6	CO4
5.	**Finding Similar Items, Clustering, and Real-Time Big Data Models:** Distance Measures: Definition of a Distance Measure, Euclidean Distances, Jaccard Distance, Cosine Distance, Edit Distance, Hamming Distance. CURE Algorithm, Stream-Computing, A Stream-Clustering Algorithm, Initializing and Merging Buckets, Answering Queries PageRank Overview, Efficient computation of PageRank: PageRank Iteration Using MapReduce, Use of Combiners to Consolidate the Result Vector. A Model for Recommendation Systems, Content-Based Recommendations, Collaborative Filtering. Social Networks as Graphs, Clustering of Social-Network Graphs, Direct Discovery of Communities in a social graph.	6	CO5

6.	**Big Data Analytics Frameworks:** Spark Framework: Introduction to GPU Computing, CUDA Programming Model, CUDA API, Simple Matrix, Multiplication in CUDA, CUDA Memory Model, Shared Memory Matrix Multiplication, Additional CUDA API Features,Spark SQL and GraphX: SQL Context, Importing and Saving data, Data frames – using SQL, GraphX overview, Creating Graph, Graph Algorithms, Spark Streaming: Overview, Errors and Recovery, Streaming Source, Streaming live data with spark.	7	CO6

Text Books	
1.	CreAnand Rajaraman and Jeff Ullman “Mining of Massive Datasets”, Cambridge University Press, 2014
2.	Alex Holmes “Hadoop in Practice”, Manning Press, Dreamtech Press.
3.	Dan Mcary and Ann Kelly — “Making Sense of NoSQL – A guide for managers and the rest of us”, Manning Press
Reference Books	
1.	Bill Franks, Taming, ”The Big Data Tidal Wave: Finding Opportunities In Huge Data Streams With Advanced Analytics”, Wiley
2.	Chuck Lam, “Hadoop in Action”, Dreamtech Press
3.	Jared Dean, “Big Data, Data Mining, and Machine Learning: Value Creation for Business Leaders and Practitioners”, Wiley India Private Limited, 2014.
4.	Jiawei Han and Micheline Kamber, “Data Mining: Concepts and Techniques”, Morgan Kaufmann Publishers, 3rd ed, 2010.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5014S
Course Title	Soft Computing
Course Type	Program Core-IV
Prerequisites: Neural network	
Course Outcomes: At the end of the course student will be able to: **CO1.** Construct intelligent systems leveraging the archetype of soft computing techniques. **CO2.** Provide the mathematical background for carrying out the optimization associated with intelligent learning algorithms. **CO3.** Design hybrid system to revise the principles of soft computing in various applications. **CO4.** Develop familiarity with current research problems and research methods in soft computing.	

	Course Contents	Hrs.	CO
1.	**Fuzzy Logic:** Introduction to fuzzy logic, Fuzzy membership functions, Operations on fuzzy sets, Fuzzy relations, Fuzzy propositions, Fuzzy implications, Fuzzy inferences, Defuzzification Techniques, Fuzzy logic controller, Advances in fuzzy logic	12	CO1, CO4
2.	**Genetic Algorithm (GA):** Concept of GA, GA Operators: Encoding, Selection, Crossover, Mutation, Advances in GA	8	CO1, CO2, CO4
3.	**Multi-Objective Optimization:** Introduction to MOO problem, Non-Pareto based approach for MOO problem, Pareto based approach for MOO problem, Advances in MOO	8	CO1, CO2, CO4
4.	**Artificial Neural Network:** Introduction to ANN, Various architectures of ANN, Learning algorithms of ANN, Advances in ANN	8	CO1, CO4
5.	**Hybrid Systems:** Integration of Neural Networks, Fuzzy Logic and Genetic Algorithms, Research orientation of soft computing techniques	4	CO1, CO3, CO4

Text Books	
1.	S.N. Sivanandam and S.N. Deepa, “Principles of Soft Computing”, Wiley India, 2007.
2.	S. Rajasekaran and G. A. Vijayalakshmi Pai, “Neural Networks, Fuzzy Logic, and Genetic Algorithms (Synthesis and Applications)”, PHI Education, 2003.
Reference Books	
1.	Fakhreddine O. Karray and Clarence De Silva, “Soft Computing and Intelligent Systems Design (Theory, Tools and Applications)”, Pearson education, 2009.
2.	Timothy J. Ross, “Fuzzy Logic with engineering applications”, John Wiley; Sons, 2016.
3.	Melanic Mitchell, “An Introduction to Genetic Algorithm”, MIT Press, 1998.
4.	Carlos A. Coello Coello, Gary B. Lamont and David A. Van Veldhuizen, “Evolutionary Algorithms for Solving Multi-Objective Problems”, 2 nd Edition, Springer, 2007.
5.	Simon Haykin, “Neural Networks and Learning Machines”, 3 rd Edition, Pearson, 2009.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5041T
Course Title	Network Administration
Course Type	Program Elective-III
Prerequisites: Computer Networks	
Course Outcomes: At the end of the course student will be able to: **CO1.** Analyze different network architectures and their features. **CO2.** Perform various network operations and maintenance activities. **CO3.** Identify the requirements of a data center network. **CO4.** Examine and analyze the network effectively. **CO5.** Design a secure network and resolve the issues.	

	Course Contents	Hrs.	CO
1.	**Network Architecture Overview:** Physical Infrastructure, Logical Design, ISO/OSI Model, Network Topologies, Data Communication and Routing, Network Addressing and Security Features..	6	CO1
2.	**Network Operations and Processes:** Monitoring, Management: Access and Audit Trail, Life Cycle, Configuration Management, Deployment Process, Documentation: Network Design and Implementation, DNS, Labeling, Support: Tools, Organizational Structure, Network Services.	7	CO2, CO4
3.	**Datacenter Networks:** Build/Rent/Outsource, Requirements, Capacity Management, Life-Cycle Management, Patch Cables, Labeling, Console Access, Workbench, Tools and Supplies, Security Aspects.	7	CO1, CO3
4.	**Network Monitoring:** Overview, Monitoring Platforms, Data Collection and Performance Analysis, Systems and Service Monitoring, SNMP, Namespaces and Nameservices, Time Management, Customer Support, Incident Report.	8	CO2, CO4
5.	**Network Maintenance:** Change Management, Scaling and Expansion, Server Upgrades, Maintenance Windows, Software Repositories and Licencing, Data Storage, Backup and Restore, Disaster Recovery.	8	CO2, CO4

6.	**Network Security:** Basic Security Measures, The OSI Security Architecture, Security Attacks, Security Services and Mechanisms, Fundamental Security Design Principles, Attack Surfaces and Attack Trees, A Model for Network Security, Standards, Legal and Ethical Issues.	6	CO1, CO5

Text Books	
1.	Thomas Limoncelli, Christina Hogan, Strata Chalup “The Practice of System and Network Administration, Volume 1”, 3ed, Addison-Wesley (2017)
2.	Evi Nemeth, Garth Snyder, Trent Hein, Ben Whaley, Dan Mackin, “UNIX and Linux System Administration Handbook”, 5ed, Addison-Wesley (2018)
Reference Books	
1.	Mark Burgess, “Principles of Network and System Administration”, 2ed, John Wiley (2004)
2.	William Stallings, “Network Security Essentials”, 6ed, Pearson (2017)

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5042T
Course Title	Ad-hoc Network
Course Type	Program Elective-III
Prerequisites: Computer Network.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Justify the wireless technology requirements. **CO2.** Apply the MAC protocols for the design of wireless network. **CO3.** Inspect the behaviour of Mobile IP and routing protocols. **CO4.** Examine the impact of transport layer protocols over wireless and wired medium. **CO5.** Evaluate resource optimization techniques for better performance. **CO6.** Design and deploy different wireless networks.	

	Course Contents	Hrs.	CO
1.	**Introduction:** Wireless technology: Signals, Antennas, Spectrum, Radio Propagation Mechanism, Characteristics of wireless Channel, Issues in Ad Hoc Wireless Networks, Multiplexing, Modulation.	4	CO1
2.	**MAC Protocols for Ad Hoc Wireless Networks:** Introduction, Issues in Designing a MAC Protocol for Ad Hoc Wireless Networks, Motivation for a specialized MAC, Design Goals of a MAC Protocol for Ad Hoc Wireless Networks, Classifications of MAC Protocols.	6	CO2
3.	**Wireless LAN and PAN:** Fundamentals of WLAN, IEEE 802.11 standards, HIPERLAN, Bluetooth. Wireless Internet: Introduction, Mobile IP, WAP.	8	CO1, CO2
4.	**Routing Protocols for Ad Hoc Wireless Networks:** Introduction, Issues in Designing a Routing Protocol for Ad Hoc Wireless Networks, Classifications of Routing Protocols: Table-Driven Routing Protocols, On-Demand Routing Protocols, Hybrid Routing Protocols, Routing protocols with efficient flooding mechanisms (OLSR), power-aware routing protocols, Multicast routing: zonal routing etc.	6	CO3

5.	**Transport Layer and Security Protocols for Ad Hoc Wireless Networks:** Introduction, Traditional TCP, indirect TCP, Snooping TCP, Fast retransmit/fast recovery, transmission/time out freezing, selective retransmission, transaction-oriented TCP. Network Security Attacks: Network Layer, Transport Layer, Application Layer, Key Management, Secure Routing in Ad Hoc Wireless Networks.	6	CO4
6.	**Quality of Service in Ad Hoc Wireless Networks:** Introduction, Issues and Challenges in Providing QoS in Ad Hoc Wireless Networks, Classifications of QoS Solutions, QoS Frameworks for Ad Hoc Wireless Networks. Energy Management in Ad Hoc Wireless Networks: Introduction, Need for Energy Management in Ad Hoc Wireless Networks, Classification of Energy Management Schemes, Battery Management Schemes, Transmission Power Management Schemes, System Power Management Schemes. Case study: Cellular network, 802.15.4, 802.16 Recent Advances in the domain.	10	CO5, CO6

Text Books	
1.	Jochen Schiller, “Mobile communications”, 2nd Edition, Pearson Education, 2008.
2.	C. Siva Ram Murthy and B.S. Manoj, “Ad Hoc Wireless Networks: Architectures and Protocols”, third Edition, Pearson education, 2008.
Reference Books	
1.	Wiiliam Stallings, “Wireless Communications and Networks” Prentice Hall, 2nd edition, 2005.
2.	C K Toh, “Ad Hoc Mobile Wireless Networks: Protocols and Systems”, 1st Edition, Pearson education, 2002.
3.	Rappaport, “Wireless Communications Principals and Practices”, 2nd Edition, Pearson Education Pvt. Ltd, 2003.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5043T
Course Title	Software Project Management
Course Type	Program Elective-III
Prerequisites: Software Engineering.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Apply project management concepts and techniques to software projects. **CO2.** Explain Process groups and Project Integration Techniques for software projects. **CO3.** Mange scope and time of software projects. **CO4.** Estimate cost and assure quality of software projects. **CO5.**Determine human resource and manage communications among project team members. **CO6.** Control risk of software projects and carry out procurement for software projects.	

	Course Contents	Hrs.	CO
1.	**Introduction to Project Management:** Introduction to Project and Project Management, Program and Project Portfolio Management, The Role of the Project Manager, A Systems View of Project Management, organizational structure, Stakeholder Management, Project Phases and the Project Life Cycle, List the project management processes.	6	CO1
2.	**The Project Management Process Groups and Project Integration Management:** Project Management Process Groups, Developing Software Project Management Methodology. Project Integration Management: Develop Project Charter, Develop Project Management Plan, Direct and Manage Project Execution, Monitor and Control Project Work, Perform Integrated Change Control, Close Project or Phase.	8	CO2

3.	**Project Scope and Time Management:** Project Scope management: Collect requirements, Define scope Create Work Breakdown Structure (WBS), Verify scope, Control scope. Project Time Management: Define activities, Sequence activities, Estimate resources and duration for each of the defined activities, Develop a project schedule, Develop a process for monitoring and controlling the project schedule.	6	CO3
4.	**Project Cost and Quality Management:** Project Cost Management: Planning Cost Management, Processes of Cost Management: Estimate Costs, Determine Budget, Control Costs. Project Quality Management: Pan for quality in your projects Perform quality assurance and quality control, Tools and Techniques for Quality Control.	6	CO4
5.	**Project Human Resource and Communications Management:** Project Human Resource Management: Developing the Human Resource Plan, Acquiring the Project Team, Developing the Project Team, Managing the Project Team. Project Communications Management: Planning Communications Management, Managing Communications, Controlling Communications, Information Distribution, Performance Reporting, Managing Stakeholders.	6	CO5
6.	**Project Risk Management and Procurement Management:** Plan risk management: Identifying Risks, Performing Qualitative Risk Analysis, Performing Quantitative Risk Analysis, Planning Risk Responses, Controlling Risks. Planning Procurement Management: Plan procurement, Conduct procurement, Administer procurement, Close procurement.	6	CO6

Text Books	
1.	Kathy Schwalbe, “Information Technology Project Management”, Revised, 7th Edition, Cengage Learning, 2014.
2.	John M. Nicholas, Herman Steyn, “Project Management for Business and Technology”, PHI, 4th Edition, 2000.
Reference Books	
1.	Joseph Phillips, “IT Project Management: On Track from Start to Finish”, Mc-Graw Hill Publication, 3rd Edition, 2010.
2.	P. Jalote, “Software Project Management in Practice”, Pearson Education, 2002.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5044T
Course Title	Software Architecture
Course Type	Program Elective- III
Prerequisites: Software Engineering, Object Oriented Programming language.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Describe the role and function of software architecture in modern development methodologies **CO2.** Identify architectural styles in existing software systems and apply to new software **CO3.** Analyze architectural patterns in existing software systems and apply to new software **CO4.** Identify and evaluate appropriate software quality attributes and requirements	

	Course Contents	Hrs.	CO
1.	**Architecture Description and Architectural Views:** Prescriptive vs Descriptive Architecture, Architectural Design- DSSA, Architectural Pattern, Architectural Styles: Layered styles, Data-flow styles, Shared memory, interpreter Style, Implicit Invocation Styles, Peer to Peer Styles. Complex Architectural Style: C2 and CORBA Connectors: Roles, Types of Connector, Data Distribution connector: Event based, Grid-based, Client-server based, P2P based.	6	CO1
2.	**Micro Architectural design principles and Patterns:** Single Responsibility Pattern, Open close principle, Liskovs substitution principle, Interface segregation principle, dependency inversion principle, Singleton Design Pattern, Factory Design Pattern, Strategy Design Pattern, Observer Design Pattern, Builder Design Pattern, Adapter Design Pattern, Decorator Design Pattern, Chain of Responsibility Design Pattern	8	CO2, CO3

3.	**Component Principles and Middleware Architectures:** Component Cohesion- reuse release equivalence principle, common closure principle, common reuse principle. database middleware, application server middleware, Message-Oriented Middleware - Publish–Subscribe, Message Brokers, Web middleware.	6	CO2, CO3
4.	**Applied Architectures and Styles, Designing for Non-Functional Properties:** Macro Architectural Patterns: Event Driven Architecture, Microkernel Architecture, MicroServices Architecture Pattern, Service-Oriented Architectures SOAP and REST	6	CO2
5.	**Software Quality Attributes:** Performance, Scalability, Modifiability, Security, Availability, Efficiency, Complexity, Heterogeneity, Adaptability, Dependability.	6	CO4
6.	**Case studies:** Case studies of Microservices-Based Architecture, SOAP and rest	6	CO2, CO3, CO4

Text Books	
1.	Richard N. Taylor, Nenad Medvidovic, Eric Dashofy, "Software Architecture: Foundations, Theory, and Practice", Addison-Wesley, 2007.
2.	Len Bass, Paul Clements, Rick Katzman, Ken Bass, "Software Architecture in Practice", Addition Wesley, 2014
Reference Books	
1.	Ian Gorton, "Essential Software Architecture", Springer 2011
2.	Robert C Martin, "Clean Architecture - a craftsman's guide to software structure and design", Pearson education 2018.
3.	Eberhard Wolff, "Microservices-Flexible Software Architecture", Pearson Education 2017.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5051S
Course Title	Network Security
Course Type	Program Elective -IV
Prerequisites: Nil.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Demonstrate the concept of cryptography, Network security ,Layered Architecture. **CO2.** Evaluate network Stack Vulnerabilities, threats and counter measures **CO3.** Analyze the network and web attacks at different layers of TCP/IP stack. **CO4.** Explore a better understanding of Network Security Protocols. **CO5.** Apply the fundamentals of security in programs, operating systems and databases.	

	Course Contents	Hrs.	CO
1.	**Overview of Security:** Motivation, Terminology/Background, Cryptography Overview, Confidentiality, Integrity, Authentication: Foundations,Symmetric key encryption, Block modes, Asymmetric key encryption, Hashes, MACs, Digital Signatures, key distribution, one way/mutual/mediated authentication,Protocols Overview,Introduction to Network security ,Layered architecture, Client Server architecture, Peer-to-Peer Architecture.	4	CO1
2.	**Vulnerabilities and Threat in Network Stack:** Basic services used and provided by TCP/IP Stack,Types of devices constituting a network. Concept of Internet Service Providers (ISPs)and overall conceptual view of the Internet. Routing fundamentals. Different types of networks such as LAN, WAN, VPN, etc, TCP/IP Protocol and its Vulnerabilities, Hyper Text Transfer Protocol (HTTP) and corresponding cyber security vulnerabilities.TCP/UDP/IP Vulnerabilities,Data link layer protocol vulnerability, Routing attacks.	7	CO2

3.	**Network and Web Attacks:** Attacks at link/network/transport/application layer,Denial of Service (DOS) attacks, Firewalls, Intrusion Detection,Malware and its types, The OWASP top 10-attacks,Client side and Server side attacks: Injection, Vulnerability, SQL injection, Cross Site Scripting (XSS) . Session Hijacking,Phishing, Click jacking, scripting, Broken Authentication and Session Management, Insecure Direct Object References, Cross Site Request Forgery (CSRF) Vulnerability, Failure to Restrict URL Access, Invalidated Redirects and Forwards.	7	CO3
4.	**Network Security protocols:** Application Layer: SSH.PGP,MIME,Transport Layer: TLS/SSL, Network Layer: IPSec, Link Layer: WPA, WEP, Open source tools for defense mechanism, Network Security controls, How to use network analysis tool: Wireshark and NMAP, ESAPI structure: security mechanism to mitigate the top 10 threats of OWASP, Defenses against the same.	7	CO4
5.	**System Security:** Introduction to System Security, Server Security, OS Security,Database Security, Various concepts of system and server security,overview of program security.	7	CO5
6.	**Advances in network security:** Cyber Security and Recent Technologies ,Security aspects in IoT, Cloud Computing and Image/video data. Bio-metrics, Mobile Computing and Hardening on android and ios, IOT Security, Android Malware Analysis, Experimentation using open source tools	7	CO5, CO1

Text Books	
1.	William Stallings, “Cryptography and Network Security”, Pearson Education/PHI, 2006
2.	Network Security Bible second edition by eric cole
Reference Books	
1.	
2.	

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5052S
Course Title	Blockchain Technology
Course Type	Program Elective-IV
Prerequisites: Cryptography, Programming language.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Identify essential features of blockchain, cryptocurrency, and distributed ledger technology. **CO2.** Apply various blockchain concepts to analyze examples, proposals, case studies, and preliminary blockchain system design discussions **CO3.** Classify consensus algorithms for different case studies. **CO4.** Design smart contract for real world applications. **CO5.** Understand and Asses security issues of blockchain and smart contracts. **CO6.** Evaluate potential uses cases from a business, legal, and engineering perspective.	

	Course Contents	Hrs.	CO
1.	**Foundations:** Ledgers - Databases - Distributed systems - Hash-linked chains, distributed ledger, Blockchain as new form of trust, distributed consensus, Double- Spending Problem, GHOST protocol.	4	CO1
2.	**Bitcoin Blockchain:** Bitcoin Blockchain and scripts, Use cases of Bitcoin Blockchain scripting language in micro-payment, escrow etc Downside of Bitcoin – mining, UTXO Model, Transactions, The Merkle Root, Signing and Validating Transactions, The Coin-base Transaction, Wallets.	8	CO1, CO2
3.	**Ethereum Blockchain:** Ethereum and Smart contracts, Lightweight wallets, Hierarchical deterministic wallets, Extended public keys, Deriving hardened private keys, Public key math, Public key multiplication, Public key encoding, Merkle trees, Security of lightweight wallets, NFTs and ERC-721 Tokens Stablecoins and other ERC-20 Tokens Decentralized Finance (DeFi), Layer 2 and Payment Channel Networks (Lightning)	8	CO2, CO3

4.	**Hyperledger Blockchain:** Fabric network: Infrastructure, Participants In Hyperledger Blockchain Network, Chaincode, Types Of Peers, Transaction Life-Cycle Of Hyperledger Fabric,	8	CO2, CO3
5.	**Privacy, Security issues in Blockchain:** Pseudo-anonymity vs. anonymity, Zcash and Zk-SNARKS for anonymity preservation, attacks on Blockchains – such as Sybil attacks, selfish mining, 51% attacks - - advent of algorand, and Sharding based consensus algorithms, User Addresses and Privacy Security issues in Blockchain: Anonymity, Sybil Attacks, Selfish Mining, 51/49 ratio Attacks	4	CO5
6.	**Case Studies:** Case studies of Blockchain: e-Governance, e-Commerce, Database Applications where third party is involved Use Cases: Cryptocurrency and Other Sectors like Finance, Land Registration, Voting System, IoT and smart cities, and Healthcare, etc. Block chain in Social Networking, block chain in 5G, Cross border payments, Know Your Customer (KYC), Food Security, Mortgage over Blockchain, Crowd funding, International Trade finance, Supply Chain Management, Smart cities, Provenance, Slock	6	CO4, CO6

Text Books	
1.	Andreas M. Antonopoulos, Gavin Wood, Dr. Gavin Wood, “Mastering Ethereum: Building Smart Contracts and DApps”, O’Reilly Media, Incorporated, 2018.
2.	A. Narayanan, J. Bonneau, E. Felten, A. Miller, and S Goldfeder, “Bitcoin and Cryptocurrency Technologies”, Princeton University Press, 2016
3.	Andreas M. Antonopoulos, “Mastering Bitcoin Programming the Open Blockchain”, O’Reilly Media 2017.
Reference Books	
1.	M. Swan, “Blockchain: Blueprint for a New Economy”, OReilly, 2015

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5053S
Course Title	Parallel Algorithms
Course Type	Program Elective-IV
Prerequisites: Discrete Mathematics , Data Structures, Design and Analysis of algorithms.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Analyze the applications for parallel algorithms and distributed processing. **CO2.** Design and analyze the computational complexity of a parallel algorithm **CO3.** Evaluate parallel concepts with different programming languages. **CO4.** Categorize various non-numerical and numerical algorithms in Parallel algorithms.	

	Course Contents	Hrs.	CO
1.	**Introduction:** Need for parallel computers, Scope of Parallel Computing, Theoretical models of parallel computation: PRAM, interconnection networks, Performance Metrics for Processors, Performance of Parallel Algorithms, Parallel Programming Models, Analyzing and Expressing parallel algorithms, Distributed processing.	6	CO1
2.	**Fundamentals of Parallel Algorithm Design:** Principles of Parallel Algorithm Design: Preliminaries, Decomposition Techniques, Characteristics of Tasks and Interactions, Mapping Techniques for Load Balancing, Methods for Containing Interaction Overheads, Parallel Algorithm Models.	6	CO2
3.	**Analytical modeling and parallel program:** Sources of Overhead in Parallel Programs, Basic performance measures, measures of data communication, Multiplication factors, effect of software, cognitive system, bench-marking, defining and measuring cost, scalability of parallel system., Asymptotic Analysis of Parallel Programs.	6	CO2

4.	**Parallel Programming Models:** Basic Communication Operations , Parallel programming with message passing using MPI, , Directives based parallel programming models, Open MP, OpenACC, Thread level parallelism, GPU architectures and the programming model for GPGPU (CUDA), Strategies for efficient CUDA programming.	8	CO3
5.	**Non-numerical algorithms:** Sorting: Hyper quick sort, Merge sort, Bitonic merge sort, odd even transposition, Enumeration sort, Graphs: Graph coloring, Minimal spanning tree, Shortest path algorithm, Searching and selection, Dynamic programming.	8	CO4
6.	**Numerical algorithms:** Dense matrix algorithms: Matrix-Vector Multiplication, Matrix-Matrix Multiplication, Matrix-Matrix Multiplication, Fast Fourier transforms: The Serial Algorithm, The Binary-Exchange Algorithm, The Transpose Algorithm, Limits to parallelizability. Advances in the domain.	6	CO4

Text Books	
1.	AnanthGrama, Anshul Gupta, George Karypis, Vipin Kumar, “Introduction to Parallel Computing, Pearson Education”, Second Edition, 2007.
2.	Fayez Gebali, “Algorithms and Parallel Computing”, Wiley publications, 2011.
Reference Books	
1.	Michael J. Quinn, “Parallel Programming in C with MPI and OpenMP”, McGraw-Hill International Editions, Computer Science Series, 2008.
2.	Barbara Chapman, “Using OpenMP - Portable Shared Memory Parallel Programming”, The MIT Press Cambridge, Massachusetts London, England 2008.
3.	David B. Kirk and Wen-mei W. Hwu, “Programming Massively Parallel Processors A Hands-on Approach”, Second Edition, Elsevier 2013.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5054S
Course Title	Software Defined Network
Course Type	Program Elective-IV
Prerequisites: Computer Network.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Demonstrate the working of SDN. **CO2.** Emulate SDN using openflow. **CO3.** Programme the SDN. **CO4.** Apply SDN in data center. **CO5.** Study SDN security and its applications.	

	Course Contents	Hrs.	CO
1.	**Computer Network Design Foundation:** Introduction to SDN: History of Software Defined Networking (SDN), Modern Data Center, Traditional Switch Architecture, Why SDN, Evolution of SDN, How SDN Works – Centralized and Distributed Control and Date Planes, The Genesis of SDN	06	CO1
2.	**Open Flow and SDN Controllers:** Open Flow Specification, SDN via APIs, SDN via Hypervisor- Based Overlays – SDN via Opening up the Device, General Concepts, OpenFlow Protocol, SDN Controllers: Introduction, VMware - Nicira - VMware/Nicira - OpenFlow-Related - Mininet - NOX/POX - Trema - Ryu - Big Switch Networks/Floodlight Layer 3 Centric - Plexxi - Cisco OnePK	06	CO2
3.	**SDN Programming:** Northbound Application Programming Interface, Current Languages and Tools, Composition of SDNs – Network Functions Virtualization (NFV) and Software Defined Networks: Concepts, Implementation and Applications, NetApp Development, Network Slicing	06	CO3

4.	**SDN in Data Center:** SDN in the Data Center - SDN in Other Environments - SDN Applications - SDN Use Cases - The Open Network Operating System 3, Multitenant and Virtualized Multitenant Data Center – SDN Solutions for the Data Center Network – VLANs – EVPN – VxLAN – NVGRE	08	CO4
5.	**SDN Security:** Security Characteristics of SDN, Security Analysis and Potential attacks in SDN,Security Principles of SDN, Solutions to the security issues in SDN, Network Security enhancement using the SDN Framework – Issues and Challenges, Threats to SDN -Networks, Controllers, Applications	06	CO5
6.	**SDN Applications and SDN Future:** SDN applications-Reactive versus Proactive Applications, Analysing Simple SDN Applications, A Simple Reactive Java Application, Using the Floodlight Controller, Using the Open Daylight Controller, Access Control for the Campus, Traffic Engineering for Service Providers, Applications of SDN to Real Networks. SDN Future -Potential Novel Applications of Open SDN-Managing Non-traditional Physical Layer Links, Applying Programming Techniques to Networks, Security Applications, Roaming in Mobile Networks, Traffic Engineering in Mobile Networks, SDN Open Source - SDN Futures - Final Thoughts and Conclusions	08	CO5

Text Books	
1.	Paul Goransson and Chuck Black, “Software Defined Networks: A Comprehensive Approach”, Morgan Kaufmann, 2014, ISBN: 9780124166752, 9780124166844.
2.	SiamakAzodolmolky, “Software Defined Networking with Open Flow, Packt Publishing, 2013, ISBN: 9781849698726.
3.	Thomas D. Nadeau, Ken Gray, “SDN: Software Defined Networks, An Authoritative Review”
Reference Books	
1.	1. Vivek Tiwari, “SDN and OpenFlow for Beginners”, Digital Services, 2013, ISBN: 10: 1-940686-00-8, 13: 978-1-940686-00-4.
2.	Fei Hu, “Network Innovation through OpenFlow and SDN: Principles and Design”, CRC Press, 2014, ISBN: 10: 1466572094.
3.	Open Networking Foundation (ONF) Documents, https://www.opennetworking.org, 2015

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5065S
Course Title	Human Computer Interaction
Course Type	Open Elective-II
Prerequisites: Nil.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Analyze the importance of various aspects of HCI. **CO2.** Apply different tools for interactive system design for HCI **CO3.** Design different web interfaces of HCI. **CO4.** Evaluate interaction using different interactive style. **CO5.** Distinguish usability aspects of software system. **CO6.** Create software system using UX and UI.	

	Course Contents	Hrs.	CO
1.	**Introduction to Human-Computer Interaction:** Objective, Overview and historical evolution of HCI, Ergonomics, Interaction styles, Elements of the WIMP (windows, icons, pointers, menus) interface, interactivity, the context of the interaction, paradigms for interaction, Cognitive walkthrough.	4	CO1
2.	**Design process:** Human interaction with computers, importance of human characteristics human consideration, Human interaction speeds, understanding business junctions. Screen Designing, Interaction Design, Interactive Design, Interface Design, GUI Design, Software Tools, Dialog Design.	8	CO2
3.	**Web Interfaces:** Designing Web Interfaces – Drag and Drop, Direct Selection, Contextual Tools, Overlays, Inlays and Virtual Pages, Process Flow	6	CO3
4.	**Interaction Styles:** Concept of combined reality, virtual reality, technologies, existing scientific and commercial projects. Sensing and tracking. Sensors for sensing of fingers, hands and touching. Interactive digital surfaces, manipulation of digital objects, displays with rear projection.	6	CO4

5.	**Usability Testing and Analytic Evaluation:** Involves usability testing through examples, the basics of experimental design, the methods used in usability testing, the role of field studies in evaluation, the important concepts associated with inspection methods, how heuristic evaluation can be adapted to evaluate different types of interactive products, what is involved in doing heuristic evaluation and various kinds of walkthrough, how to perform predictive technique, and when to use them, the advantages and disadvantages of using analytical evaluation.	8	CO5
6.	**Design Case Studies:** 1] Multikey press Hindi Text Input Method on a Mobile Phone, 2] GUI design for a mobile phone based Matrimonial application. 3] Employment Information System for unorganized construction workers on a Mobile Phone	6	CO6

Text Books	
1.	J. Preece, Y. Rogers and H. Sharp, "Interaction design: Beyond Human-Computer Interaction", 4/e, John Wiley and Sons, 2002.
2.	Shneiderman B., Plaisant C., Coen M., Jacobs S., "Designing the User Interface: Strategies for Effective Human-Computer Interaction", 5/e Pearson Education, 2009.
Reference Books	
1.	Alan Dix, Janet Fincay, Gre Goryd, Abowd, Russell Bealg, "Human – Computer Interaction. Pearson Education, 2003.
2.	Helen Sharp, Jennifer Preece, Yvonne Rogers, "Interaction Design", Wiley Dreamtech, 2019.
3.	Soren Lauesen, "User Interface Design", Pearson Education, 2005.
4.	Bill Scott and Theresa Neil, "Designing Web Interfaces", O'Reilly, 2009.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5066S
Course Title	Machine Learning
Course Type	Open Elective-II
Prerequisites: Basic understanding of probability and statistics, linear algebra and calculus.	
Course Outcomes: At the end of the course student will be able to: **CO1.** . Gain knowledge about basic concepts of Machine Learning **CO2.** Identify machine learning tools and techniques solving real time problems **CO3.** Solve the problems using various machine learning techniques **CO4.** Optimise and test the model for best performance. **CO5.** Exploring the advances in machine learning future to solve real time case studies	

	Course Contents	Hrs.	CO
1.	**Introduction:** Introduction and Basic Concepts of ML, Taxonomy of ML, Types of machine learning: Supervised Learning, Regression Vs Classification, Unsupervised Learning, Clustering, Classification, Rules mining, Prediction, Issues in machine learning.	6	CO1
2.	**Machine learning Tools:** R, Python, Scikit Learn, BigML , WEKA, or. any one platform to make machine learning in practice with case studies.Data and Data understanding, Data pre-processing. Learning Association Rules: Mining Frequent Patterns, Apriori algorithm, and other varients of Association rules mining algorithms.	8	CO2, CO3
3.	**Supervised Learning:** Decision Trees: ID3, Classification and Regression Trees, Regression. Neural Networks, Support vector machines, Generalized Linear Models (GLM), Probabilistic Learning: Bayesian Learning, Bayes Optimal Classifier, Naıve Bayes Classifier, Markov Decision Process (MDP). Ensemble Learning: Model Combination Schemes, Bagging: Random Forest Trees, Boosting: Adaboost, Stacking	08	CO1, CO3

4.	**Unsupervised learning:** Clustering, Instance-based learning, K-nearest Neighbour, Dimensionality Reduction, K-Mode Clustering, Expectation Maximization, Gaussian Mixture Models.	6	CO3
5.	**Balanced Machine Learning Model and Model Evaluation:** What Are Evaluation Metrics? Types of Predictive Models, Confusion Matrix, F-Score, Accuracy, Precision, Recall, Gain and Lift Charts , Kolmogorov-Smirnov Chart , Area Under the ROC Curve, Log Loss, Gini Coefficient, Concordant – Discordant Ratio, Root Mean Squared Error (RMSE), Root Mean Squared Logarithmic Error, R-Squared/Adjusted R-Squared, Cross Validation, Bias-Variance and Error Analysis, Bias/variance trade-off, Error Analysis, Normal Equations, Variance, Gradient Descent, Model Balancing: Overfitting, underfitting, Variance, Bias and Model Complexity in Machine Learning.	8	CO4
6.	**Introduction to Advanced topics in Machine Learning:** Deep Neural Networks, Vectorization, Backpropagation, Forward propagation, multi-label classification, Conditional Random Fields (CRFs), Reinforcement Learning, Spectral clustering- Semi-supervised learning. Deep Learning Models: Introduction to NN, important terms in NN, DNN, CNN, RNN, Model Training and testing, Pretrain models, parameter tuning and customized models, Deep learning in images processing, video processing, text processing.	6	CO5

Text Books	
1.	Ethem Alpaydin, “Introduction to Machine Learning”, MIT Press, Prentice Hall of India, Third Edition, 2014.
2.	Miroslav Kubat, “An Introduction to Machine Learning”, Springer, 2015.
Reference Books	
1.	Tom Mitchell, “Machine Learning”, McGraw-Hill, 2017
2.	Aurélien Géron, “Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems”, Third Edition, OReilly Publication, 2022
3.	John D. Kelleher, Deep Learning, The MIT Press Essential Knowledge series, 2019
4.	Jerome Friedman, Robert Tibshirani, Trevor Hastie, “The Elements of Statistical Learning” Springer, 2017.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5074L
Course Title	Laboratory-4: Devops
Course Type	Laboratory

Prerequisites: Software Engineering, Web Development, Cloud, Networking.

Course Outcomes: At the end of the course student will be able to:
CO1. Track the changes in the software application development and maintenance.
CO2. Demonstrate continuous integration and development of the software application.
CO3. Design continuous deployment strategy for the software application.
CO4. Apply test automation to the software application.
CO5. Perform configuration management and monitoring of the software application.
CO6. Develop and deploy a software application in the cloud environment.

It is expected to conduct 8-10 experiments in the field of knowledge. One hour shall be tutored about theory and relevant tools to the students, and students shall perform the experiment. This is continuous evaluation, hence all experiments shall be evaluated in the same week. A sample list of experiments is given it may be altered as and when required.

Sr. No.	Course Contents
1.	Practice source code management using Git and GitHub.
2.	Jenkins installation and setup, explore the environment. Demonstrate continuous integration and development using Jenkins.
3.	Explore Docker commands for content management. Develop a simple containerized application using Docker.
4.	Integrate Kubernetes and Docker. Automate the process of running containerized applications developed in exercise 3 using Kubernetes.

5.	Install and Explore Selenium for automated testing. Write a simple program in JavaScript and perform testing using Selenium.
6.	Install Ansible in the VM. Setup the inventory, run Ansible AdHoc commands. Also run a playbook in Ansible.
7.	Monitor systems, networks and infrastructure using Nagios application.
8.	Design an application using Cloud Services to apply DevOps concepts.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5075L
Course Title	Laboratory 5: Deep learning
Course Type	Laboratory
Prerequisites: Neural network, Python, NumPy, TensorFlow, Keras.	
Course Outcomes: At the end of the course student will be able to: **CO1.** Build and train the deep learning model for various application. **CO2.** Identify the key deep learning architecture parameters. **CO3.** Use best practices to train and develop test sets and analyze bias/variance for building DL applications. **CO4.** Apply test automation to the software application. **CO5.** Model the character-level language, natural language processing, etc. Architectures.	

It is expected to conduct 8-10 experiments in the field of knowledge. One hour shall be tutored about theory and relevant tools to the students, and students shall perform the experiment. This is continuous evaluation, hence all experiments shall be evaluated in the same week. A sample list of experiments is given it may be altered as and when required. **Reference online course:** Andrew NG, “Deep Learning Specialization”, Coursera.

Sr. No.	Course Contents
1.	Write a programme to build the deep neural network using NumPy.
2.	Write a programme to regularization in the deep learning model to handle the over fitting and also, compare the various optimization methods that can speed up learning and parameter optimization using NumPy.
3.	Write a programme to build the deep neural network using tensorFlow and perform the hyperparamter tuning, regularization, and optimization.
4.	Write a programme to build ConvNet in TensorFlow for a classification problem.
5.	Write a programme to build ConvNet using Transfer Learning approach.
6.	Write a programme to perform the Neural Style Transfer algorithm.

7.	Write a programme to build recurrent nueral network for text data using TensorFlow.
8.	Write a programme to build long-short-term-memory for sequential data using TensorFlow.
9.	Write a programme to use word vector representations to build an Emojifier for finding the most appropriate emoji to be used with this sentence.
10.	Write a programme to build neural machine translation.
11.	Write a programme to perform the name entity recognition.
12.	Write a programme to perform the trigger word/keyword detection from speech data.

Programme Name	M. Tech. Computer Engineering
Course Code	COCE5076L
Course Title	Laboratory-6: Big Data Analytics
Course Type	Laboratory

Prerequisites: Database Management System, Machine Learning, Computer Network.

Course Outcomes: At the end of the course student will be able to:
CO1. Perform regression analysis ,clustering , reading big data from online website
CO2. Perform setup of hadoop cluster and illustrate hdfs commands.
CO3. Design the Map reduce algorithms to perform number processing on single node cluster.
CO4. Apply the map reduce algorithms to perform analytics on single node cluster.
CO5. Solve the database operations using Mongo DB.
CO6. Illustrate the performance of hive, pig, spark, scala programs.

It is expected to conduct 8-10 experiments in the field of knowledge. One hour shall be tutored about theory and relevant tools to the students, and students shall perform the experiment. This is continuous evaluation, hence all experiments shall be evaluated in the same week. A sample list of experiments is given it may be altered as and when required. **Reference online course:** Andrew NG, “Deep Learning Specialization”, Coursera.

Sr. No.	Course Contents
1.	Setup Hadoop Single Node cluster and compare Hadoop 1.x, Hadoop 2.x and Hadoop 3.x
2.	Setup Hadoop multimode cluster and compare the execution time for single node and multimode cluster on a input file size greater than 4.9 GB.
3.	Write a MapReduce code which prints words with a) frequency three only. b) The same set of integers, but with each integer appearing only once. c) The count of the number of distinct integers in the input. d) Aggregate inputs and search records based on aggregation output.
4.	Setup HIVE and Pig and compare their architecture.
5.	Setup HBase and study its architecture.

6.	Setup Spark and write Scala code for wordcount.
7.	Write Scala code for counting all lowercase characters in an input file.
8.	Implement Kmeans clutering algorithm using Spark MLib.
9.	Implement streaming using Spark streaming and Apache Kafka.
10.	Write a programme to perform the trigger word/keyword detection from speech data.