Sardar Patel Institute of Technology
Bhavan's Campus, Munshi Nagar, Andheri (West), Mumbai-400058-India
(Autonomous Institute Affiliated to University of Mumbai)
Department of Computer Engineering

M.Tech. (Computer Engineering) Syllabus (Semester I-II)

Semester-I

Course (Category) Code	Course Name	Teaching Scheme (Hrs/week)					Credits Assigned			
		L	T	P	O	E	L	T	P	Total
(PC)	Advanced Algorithms and Complexity	3	0	0	5	8	3	0	0	3
CS501		Examination Scheme								
		Component		ISE		MSE		ESE		Total
		Theory		75		75		150		300
		Laboratory		-		-		-		-

Pre-requisite Course Codes, if any.	1. Data Structures and Algorithms 2. Programming Language
Course Objective:	
Course Outcomes (CO):*At the End of the course students will be able to*	
CS501.1	Analyze the asymptotic performance of algorithms.
CS501.2	Apply the approach of dynamic programming to solve and analyze engineering problems and Perform amortized analysis of algorithms
CS501.3	Apply various graph algorithms to solve and analyze engineering Problems.
CS501.4	Apply and prove NP completeness of given problem and solve it using approximation algorithms.
CS501.5	Understand and Apply Linear, parallel and randomized algorithms to solve the engineering problems.

CO-PO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PO1	PO2	PO3
CS501.1			
CS501.2			
CS501.3			
CS501.4			
CS501.5			

CO-PEO/PSO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PEO1	PEO2
CS501.1		
CS501.2		
CS501.3		
CS501.4		
CS501.5		

BLOOM'S Levels Targeted (Pl. Tick appropriate)

Remember	Understand	Apply	Analyze	Evaluate	Create

Theory Component

Module No.	Unit No.	Topics	Ref.	Hrs.
1	Title	**Introduction**		8
	1.1	The role of Algorithms in computing, Analyzing algorithms, Designing Algorithms, Analysis of Insertion Sort, Merge sort, and Quick sort.	1	
	1.2	Growth of Functions-Asymptotic notation, Mathematical Background for algorithm analysis, Recurrences, The substitution method, The recursion-tree method, The master method, and Proof.	1	
2	Title	**Programing Strategies: Dynamic Programing and Amortized Analysis**		10
	2.1	**Dynamic Programing:** Elements of dynamic programming, Assembly Line Scheduling, Matrix-chain multiplication, Longest common subsequence. Optimal Binary search Tree	1	
	2.2	**Amortized Analysis:** Aggregate analysis, The accounting method, Table Doubling, The potential method	1	
3	Title	**Graphs algorithms and NP Completeness**		8
	3.1	**Graphs algorithms**: Single-Source Shortest Paths-The Bellman-Ford algorithm, Dijkstra's algorithm, All-Pairs Shortest Paths-The Floyd-Warshall algorithm, Maximum Flow-Flow networks, The Ford-Fulkerson method, Maximum bipartite matching,	1,2	
	3.2	**NP-Completeness:** Polynomial time, Polynomial time verification, NP-completeness and reducibility, NP-complete problems	1,2	
4	Title	**Linear Programming**		8
	4.1	**Linear Programming:** Standard and Slack Forms, Formulation, Graphical Solution, Simplex Method, Two phase, duality,	1,5	
	4.2	Sensitivity analysis, transportation and assignment problems.	1,5	
5	Title	**Parallel and Randomized Algorithms**		8
	5.1	**Parallel Algorithms:** Analysis, Models, Structures Design Techniques, Dynamics Multithreading, Greedy Scheduler, Multithreaded algorithms, Cache oblivious algorithm**.**	1,2.3	
	5.2	**Randomized algorithms:** Las Vegas and Monte Carlo, Markov Chain Model	3	
6	**Self Study**	Red Black Tree, Greedy Algorithms, Approximation algorithms, Number Theoretic Algorithms	1,2,4	**8***

Total (* Not Included)	**42**

Text Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Introduction to Algorithms	Second	Cormen, Thomas, Charles Leiserson, Ronald Rivest, and Clifford Stein	MIT Press	2009
2	Fundamentals of Computer Algorithms	Second	Horowitz, Sahani and Rajsekaran	Galgotia	2010
3	Randomized Algorithm	Second	Rajeev Motwani, Prabhakar Raghavan	Cambridge University Press	2013
4	Approximation Algorithms	Second	Vijay V. Vajirani	Springer-Verlag Berlin Heidelberg GmbH	2002

Reference Books

Sr. No	Title	Edition	Authors	Publisher	Year
5	Operations Research	Seventh	P K Gupta, D S Hira	Sultan Chand & Sons	2018
6	The Design and analysis of algorithms	Second	Aho, Hopcroft, Ullman	Pearson Education.	2006

Course (Category) Code	Course Name	Teaching Scheme (Hrs/week)					Credits Assigned			
		L	T	P	O	E	L	T	P	Total
(PC)	Information and System Security	3	0	0	5	8	3	0	0	3
		Examination Scheme								
		Component		ISE		MSE		ESE		Total
CS502		Theory		75		75		150		300
		Laboratory		--		--		--		--

Pre-requisite Course Codes, if any.	1. Operating Systems 2. Computer Communications and Networks 3. Discrete Structures and Graph Theory
Course Objective: -	
Course Outcomes (CO):*At the End of the course students will be able to*	
CS502.1	Provide the basic results of computer security and its limitations.
CS502.2	Contrast the different types of security policies, standards and practices.
CS502.3	Contrast the different cryptographic algorithms and typical applications.
CS502.4	Enforce security policies, standards and practices to a system.

CO-PO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PO1	PO2	PO3
CS502.1			
CS502.2			
CS502.3			
CS502.4			

CO-PEO/PSO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PEO1	PEO2
CS502.1		
CS502.2		
CS502.3		
CS502.4		

BLOOM'S Levels Targeted (Pl. Tick appropriate)

Remember	Understand	Apply	Analyze	Evaluate	Create

Theory Component

Module No.	Unit No.	Topics	Ref.	Hrs.
1	**Title**	**Overview of Information Security**	1,2	8
	1.1	**Introduction** - Basic Components, Threats, Policy and Mechanism, Assumptions and Trust, Assurance, Operational and Human Issues		
	1.2	**Access Control Matrix** - Protection State, Access Control Matrix Model, Protection State Transitions, Copy and Own Right, Principle of Attenuation of Privilege.		
	1.3	**Foundation Results** - General Security Question, Take-Grant Protection Model, Schematic Protection Model.		
2	**Title**	**Security Policies - I**	1,2	8
	2.1	**Security Policy Basics** - Types, Role of Trust, Types of Access Control, Policy Languages.		
	2.2	**Confidentiality Policies** - Goals, The Bell-LaPadula Model, Tranquility.		
	2.3	**Integrity Policies** – Goals, Biba Integrity Model, Lipner's Integrity Matrix Model, Clark-Wilson Integrity Model.		
3	**Title**	**Security Policies - II**	1,2	9
	3.1	**Availability Policies** – Goals, Deadlock, Denial of Service Models, Availability and Network Flooding Example.		
	3.2	**Hybrid Policies** - Chinese Wall Model, Clinical Information Systems Security Policy, Originator Controlled Access Control, Role-Based Access Control.		
	3.3	**Noninterference and Policy Composition** - The Problem, Deterministic Noninterference, Non-deducibility, Generalized Noninterference. Restrictiveness.		
4	**Title**	**Cryptosystems**	1,2	9
	4.1	**Cryptography** - Cryptanalysis, Symmetric Cryptosystems, Public Key Cryptography, Cryptographic Checksums, Digital Signatures.		
	4.2	**Key Management** - Session and Interchange Keys, Key Exchange, Key Generation, Cryptographic Key Infrastructures, Storing and Revoking Keys.		
	4.3	**Cipher Techniques** – Problems, Stream and Block Ciphers, Authenticated Encryption, Networks and Cryptography.		
5	**Title**	**System Security Mechanisms**	1,2	8
	5.1	**Access Control Mechanisms** - Eight Secure Design Principles, Access Control Lists, Capabilities, Locks and Keys, Ring-Based Access Control, Propagated Access Control Lists.		
	5.2	**Information Flow** – Basics, Non-lattice Information Flow Policies, Static Mechanisms, Dynamic Mechanisms, Integrity Mechanisms.		

	5.3	**Confinement Problem** - The Confinement Problem, Isolation, Detection of Covert Channels		
6*	**Self Study**	a) Study the assurance in requirement definition and analysis. b) Study the assurance during system and software design. c) Study the assurance in implementation and integration.		
			Total (* Not Included)	**42**

Text Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Computer Security: Art and Science	Second	Matt Bishop, Elisabeth Sullivan & Michelle Ruppel	Addison-Wesley Professional	2018
2	Introduction to Computer Security	First	Matt Bishop	Addison-Wesley Professional	2005

Reference Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Computer Security: Principles and Practice	Fourth	William Stallings & Lawrie Brown	Pearson Education	2018
2	Cryptography and Network Security Principles and Practices	Fourth	William Stallings	Addison-Wesley Professional	2005

Course (Category) Code	Course Name	Teaching Scheme (Hrs/week)					Credits Assigned			
		L	T	P	O	E	L	T	P	Total
(PC)	(A) Database Engineering	2	0	2	4	8	2	0	1	3
		Examination Scheme								
CS503A		Component		ISE		MSE		ESE		Total
		Theory		50		50		100		200
		Laboratory		50		--		50		100

Pre-requisite Course Codes, if any.	
Course Objective:	
Course Outcomes (CO): *At the End of the course students will be able to*	
CS503A.1	Compile relational database features and design of object oriented Database
CS503A.2	Examine various storage structure of large data
CS503A.3	Design Distributed databases
CS503A.4	Explore advance database systems in various domains

CO-PO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PO1	PO2	PO3
CS503A.1			
CS503A.2			
CS503A.3			
CS503A.4			

CO-PEO/PSO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PEO1	PEO2
CS503A.1		
CS503A.2		
CS503A.3		
CS503A.4		

BLOOM'S Levels Targeted (Pl. Tick appropriate)

Remember	Understand	Apply	Analyze	Evaluate	Create

Theory Component

Module No.	Unit No.	Topics	Ref.	Hrs.
1	**Title**	**Data Storage, Querying and Object Relational Databases**		
	1.1	Overview of Database: Schema and storage, Types of Single Level Ordered Indexes, Primary Index, Cluster Index, Relational schema modeling ER- EER, Object Management Group standard's UML for user interaction and functional Module Concepts of object Databases, Object Identity, Object structure, Type Constructors, Complex Data Types, Structured types and inheritance in SQL, Table Inheritance, Array and Multiset Types in SQL	1	7
	1.2	Implementing O-R Features, Object Relational Mapping, Object-Oriented versus Object Relational		
2	**Title**	**Big Data Storage Systems**		
	2.1	Introduction to Big Data management concepts such as: distributed and scalable data storage, including distributed file systems, key value stores, Hadoop Data Storage and Analysis, Comparison with other systems, Rational Database Management System, HDFS Design	3	7
	2.2	Column stores and graph databases, No SQL Storage Architecture, Interfacing and Interacting with NoSQL, Managing NoSQL Storage, Creating, Updating and Deleting Documents, Batch Insert, Remove Speed, Document Replacement, Using Modifiers, Updating Multiple Documents, Returning Updated Documents	4	
3	**Title**	**Distributed Databases**		
	3.1	Distributed Database System (DDBS), Architectures, Features of DDBS, Design issue in DDBS, Advantages, disadvantages, transparencies in DDBMS, three tier architecture case study,	2	7
	3.2	Objective of data distribution, Data Fragmentation, the allocation of Fragments, replication, Locking-Based Concurrency Control Algorithms, Introduction to distributed deadlock, Deadlock avoidance, prevention, detection and recovery, 2 phase and 3 phase commit protocol		
4	**Title**	**Emerging Databases**		
	4.1	Mobile databases, Temporal databases, Spatial databases, Mobile Databases, Geographic Information Systems, Genome Database Systems	1	7
	4.2	Multimedia Databases : Image Databases, Video Databases, Audio Databases, Block Chain Databases		
5	**Self Study**	Cassandra Installation, Cassandra key space operations, table operations and CURD operations.		6*
			Total (* Not included)	**28**

Laboratory Component (Minimum 10 Laboratory experiments are expected)

Sr. No	Title of the Experiment
1	Create an index on a database and analyze the effect of indexing over the database for different operation on it
2	PL/SQL Programming: Variables, Identifiers, Comment, PL/SQL Block Structure IF Statements: Simple IF Statements, Compound IF Statements IF-THEN-ELSE Statements Loop: Basic Loop, WHILE Loop, FOR Loop Cursor: Types of Cursor, Explicit Cursor Life Cycle, Explicit Cursor Attributes Trigger: Trigger,
3	PL/SQL Functions: Create Function, Function with Arguments, Executing Function, Dropping Function Procedures: Block Structure of Subprogram, Types of Subprograms, Procedure with Parameters, Executing Procedures, Dropping Procedures.
4	Packages: Package Specification, Package Body, Creating Package, Execution, Dropping Package
5	Design a distributed database by applying the concept of fragmentation
6	Demonstrate the concept of the 2-phase commit protocol
7	Demonstrate the concept of deadlock in distributed systems
8	Simulating Block Chain Transaction
9	Application Development for any case study / scenario(analysis design)
10	Develop a selected application with various features.

Text Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Database System Concepts	Seventh	Henry F Korth, Abraham Silberschatz and S. Sudharshan	McGraw Hill	2019
2	Principles of Distributed Database Systems	Third	M. Tamer Ozsu, Patrick Valduriez	Pearson Publication	2010
3	Fundamentals of database Systems	Seventh	Elmasri and Navathe	Pearson Education	2017

Reference Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Advanced Database Systems	First	Carlo Zanalio, StefenoCeri, Christo Faloutsos, V. S. Subrahmanaian, Roborto Zicari, Richard T. Snodgrass	Morgan Kaufmann Publishers	1997
2	SQL & NoSQL Databases: Models, Languages, Consistency Options and Architectures for Big Data Management	First	Andreas Meier, Michael Kaufmann	Springer Vieweg	2019
3	Hadoop The Definitive Guide	Fifth	Tom White	O'Reilly	2013
4	MongoDB The Definitive Guide	Third	Kristina Chodorow	O'Reilly	2015
5	SQL and PL/SQL for Oracle 11g Black Book	Second	Dr. P.S. Deshpande	DreamTech Press	2018

Course (Category) Code	Course Name	Teaching Scheme (Hrs/week)					Credits Assigned			
		L	T	P	O	E	L	T	P	Total
PC	(B) Soft Computing	2	0	2	5	09	2	0	1	3
		Examination Scheme								
		Component		ISE		MSE		ESE		Total
CS503B		Theory		50		50		100		200
		Laboratory		50		--		50		100

Pre-requisite Course Codes, if any.	Basic knowledge of Mathematics
Course Objective: To introduce soft computing concepts and techniques and foster their abilities in designing appropriate technique for a given scenario. To give students knowledge of non-traditional technologies and fundamentals of artificial neural networks, fuzzy sets, fuzzy logic, genetic algorithms. To implement soft computing based solutions for real-world problems.	
Course Outcomes (CO):*At the End of the course students will be able to*	
CS503B.1	Identify and describe soft computing techniques and their roles in building intelligent Machines.
CS503B.2	Apply fuzzy logic and reasoning to handle uncertainty and solve various engineering problems.
CS503B .3	Analyze optimization issues using Genetic Algorithm.
CS503B .4	Evaluate and compare solutions by various soft computing approaches for a given problem.

CO-PO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PO1	PO2	PO3
CS503B.1	3	-	-
CS503B.2	3	-	2
CS503B .3	3	-	-
CS503B .4	3	3	3

CO-PEO/PSO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PEO1	PEO2	PSO1	PSO2
CS503B.1	2	2	3	-
CS503B.2	2	2	3	-
CS503B .3	2	2	3	-
CS503B .4	2	2	3	-

BLOOM'S Levels Targeted (Pl. Tick appropriate)√

Remember	Understand	Apply	Analyze	Evaluate√	Create√

Theory Component

Module No.	Unit No.	Topics	Ref.	Hrs.
1	**Title**	**Introduction To Soft Computing:**	1,2	**04**
	1.1	Evolution of Computing: Soft Computing Constituents, From Conventional AI to Computational Intelligence		
	1.2	Difference between Hard and Soft computing.		
2	**Title**	**Neural Networks:**	1,2	**10**
	2.1	Fundamental concepts of Neuron model, Neural Networks Architecture, Learning rules, Activation functions, Supervised, Unsupervised and Reinforced Learning.		
	2.2	Supervised Learning Neural Network: Back-propagation Network, Radial Basis Function Network.		
	2.3	Unsupervised Learning Neural Network: Adaptive Resonance Architecture.		
3	**Title**	**Fuzzy Logic :**	2,3	**10**
	3.1	Fuzzy Sets, Operations on Fuzzy Sets, Fuzzy Relations,		
	3.2	Membership Functions: Fuzzy Rules and Fuzzy Reasoning, Fuzzy Inference Systems.		
	3.3	Design of Fuzzy Control Systems.		
4	**Title**	Genetic Algorithm:	2,3	**04**
	4.1	Introduction to Genetic Algorithm, Working Principle		
	4.2	Various Encoding methods, Fitness function.		
5	**Self Study**	Recent Trends in soft computing and deep learning, various classifiers, neural networks and genetic algorithm. Implementation of recently proposed soft computing techniques.		
			Total	**28**

Laboratory Component, if any (Minimum 10 Laboratory experiments are expected)

Sr. No	Title of the Experiment	Marks
1	Implement supervised learning neural network	5
2	Implement unsupervised learning neural network	5
3	Implement Fuzzy logic operations for a given problem.	5
4	Implement Fuzzy control system design for a given problem.	5
5	Implement Genetic Algorithm for a given problem.	5
6	Mini Project: Based on the syllabus topics ,above experiments and any other recent topic from soft computing area. Evaluation of Mini-Project will be based on the following rubrics : Performance Indicator: Marks Identification of problem: 5 Implementation: 10 Result interpretation: 5 Presentation and Report Writing: 5	25

Text Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Introduction to Artificial Neural Systems	-	Jacek M. Zurada	PWS Publishing Company	**1995**
2	Principles of Soft Computing	**III**	S.N.Sivanandam, S.N.Deepa	Wiley Publication	**2018**
3	Neural Networks, Fuzzy Logic and Genetic Algorithms	-	S.Rajasekaran, G. A. Vijayalakshami	PHI Learning Pvt.Ltd.	**2011**

Reference Books

Sr. No	Title	Edition	Authors	Publisher	Year
4	Neural Networks: A Comprehensive Foundation	-	Simon Haykin	Macmillan College Publishing Company	**1994**
5	Neural Network Design	**II**	Martin Hagan	CENGAGE Learning, India Edition	**2014**
6	Fuzzy Sets and Fuzzy Logic:Theory and Applications	-	George J. Klir and Bo Yuan	Prentice Hall	**1997**

Course (Category) Code	Course Name	Teaching Scheme (Hrs/week)					Credits Assigned			
		L	T	P	O	E	L	T	P	Total
PE	Artificial and Computational Intelligence	2	0	2	5	9	2	0	1	3
CS511		Examination Scheme								
		Component	ISE		MSE		ESE			Total
		Theory	50		50		100			200
		Laboratory	50		--		50			100

Pre-requisite Course Codes		Basics of Mathematics and statistics
At the End of the course students will be able to		
Course Outcomes	CS511.1	Understand the scope and basic components of Artificial Intelligence and Computational Intelligence
	CS511.2	Demonstrate AI problem solving with search concepts.
	CS511.3	Organize the knowledge base for reasoning and decision making.
	CS511.4	To apply computational intelligence technique to solve real world problems

CO-PO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PO1	PO2	PO3
CS511.1	3	-	-
CS511.2	3	-	2
CS511.3	3	-	-
CS511.4	3	3	3

CO-PEO/PSO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PEO1	PEO2	PSO1	PSO2
CS511.1	2	2	3	-
CS511.2	2	2	3	-
CS511.3	2	2	3	-
CS511.4	2	2	3	-

BLOOM'S Levels Targeted (Pl. Tick appropriate)√

Remember	Understand	Apply	Analyze	Evaluate√	Create√

Module No.	Unit No.	Topics	Ref.	Hrs.
1		**Introduction to Artificial Intelligence(AI) and Computational Intelligence(CI)**		5
	1.1	Overview: Definition of AI, History and Future of AI. Introduction to Computational Intelligence, Difference between AI and CI, Components of AI and CI.	1-6	
2		**Intelligent Agents and Environment**		6
	2.1	Introduction , Intelligent Systems , The Concept of Rationality , Types of Agents ,structure of agent, Environments and Its Properties , PEAS Representation for an Agent , Intelligent Agent Application ,Introduction to Expert system design	1-6	

3		**Problem-Solving and Searching**		**7**
	3.1	Introduction to Problem Solving, Problem Formulation, State–Space Representation, Difference between Conventional Problems and AI Problems.	1-6	
	3.2	Searching, Problem Characteristics and Issues in the Design of Search Programs, Types of Search Strategies.	1-6	
4		**Knowledge representation, reasoning**		**6**
	4.1	A Knowledge-Based Agent, The Wumpus World, Knowledge Representation Issues.	1-5	
	4.2	Representing and Reasoning with Uncertain Knowledge: probability, independence, Bayes rule, bayesian networks, probabilistic inference, Forward and Backward Reasoning, Matching Rule.	1-5	
5		**Statistics in Computational Intelligence**		**4**
	5.1	Calculation of standard deviation, root mean square, mean absolute error etc for measuring the fitness of a model	1-6	
6*	**Self Study**	Linear algebra fundamentals, Probability and statistics ,matrices, Prolog programming basics, R-programming		
			Total (* Not Included)	**28**

References:

1. Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, Prentice-Hall.
2. James M. Keller, Derong Liu, David B. Fogel, Fundamentals of Computational Intelligence: Neural Networks, Fuzzy Systems, and Evolutionary Computation, IEEE Press series on Computational Intelligence, Wiley Publication, July 2016.
3. Nilakshi Jain ,Artificial Intelligence : Making a System Intelligent Wiley Publication ,July 2019
4. Elaine Rich & Kevin Knight, "Artificial Intelligence", McGraw-Hill Science/Engineering/Math; 2nd edition
5. Nilsson, N.J., "Artificial Intelligence, a New Approach", Morgan Kaufmann, 2000.
6. N.P.Padhy, “Artificial Intelligence and Intelligent Systems”, Oxford University Press.

Laboratory Component, if any (Minimum 10 Laboratory experiments are expected)

Sr. No.	Title of the Experiment	Marks
1	Implement any real time problem based on Intelligent agent :	5
2	Implement any real time problem using searching	5
3	Implement any real time problem using Bayes rules	5
4	To design and implement an expert system, incorporating the match algorithm and the rule language. 1. It should provide a fact base updating function. 2. It should provide a function that check the rules’ LHS and return which rules were matched. 3. It should support firing RHS according to matches.	5
5	Implement any one computational statistical method for calculating loss /error	5
6	Mini Project: Based on the syllabus topics, above experiments and any other recent topic from ACI area. Evaluation of Mini-Project will be based on the following rubrics :	25

	<table><tr><td>Performance Indicator</td><td>Marks</td></tr><tr><td>Identification of problem</td><td>5</td></tr><tr><td>Implementation</td><td>10</td></tr><tr><td>Result interpretation</td><td>5</td></tr><tr><td>Presentation and Report Writing</td><td>5</td></tr></table>	

Text Books

Sr. No.	Title	Edition	Authors	Publisher	Year
1	Artificial Intelligence: A Modern Approach	-	Stuart Russell and Peter Norvig	Prentice-Hall	-
2	Fundamentals of Computational Intelligence: Neural Networks, Fuzzy Systems, and Evolutionary Computation, IEEE Press	-	S James M. Keller, Derong Liu, David B. Fogel	IEEE Press series	**2016**
3	Artificial Intelligence : Making a System Intelligent	**I**	Nilakshi Jain	Wiley publication	**2019**

Reference Books

Sr.	Title	Edition	Authors	Publisher	Year
4					
5	Artificial Intelligence and Intelligent Systems	-	N P Padhy	Oxford University Press	-

Course (Category) Code	Course Name	Teaching Scheme (Hrs/week)					Credits Assigned			
		L	T	P	O	E	L	T	P	Total
(PE)	Fundamentals of Data Science	2	0	2	3	7	2	0	1	3
		Examination Scheme								
		Component	ISE		MSE		ESE			Total
CS521		Theory	50		50		100			200
		Laboratory	50		--		50			100

Pre-requisite Course Codes, if any.	
Course Objective: The course is designed to provide conceptual understanding of Data Science	
Course Outcomes (CO):*At the End of the course students will be able to*	
CS521.1	To Examine applicability of statistical and probability and linear algebra within Data Science.
CS521.2	To analyze the predictive and descriptive models.
CS521.3	To experiment with different data handling operations through modeling.
CS521.4	To build a processing pipeline for Data Science analysis.

CO-PO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PO1	PO2	PO3
CS521.1			
CS521.2			
CS521.3			
CS521.4			

CO-PEO/PSO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PEO1	PEO2
CS521.1		
CS521.2		
CS521.3		
CS521.4		

BLOOM'S Levels Targeted (Pl. Tick appropriate)

Remember	Understand	Apply	Analyze	Evaluate	Create

Theory Component

Module No.	Unit No.	Topics	Ref	Hrs.
1	**Title**	**Introduction to Data Science**	1.2	6
	1.1	The Data Science process, CRISP-DM Framework, Defining goal, retrieving data, pre-processing data, exploratory data analysis, model building and data visualization, Ethical issues in data science, Difference between AI and Data Science		
	1.2	Data Science project Pipeline		
2	**Title**	Mathematical Preliminaries for Data Science	1.2	6
	2.1	Statistical Analysis: Mean, Median, Mode, Regression coefficient ,IQR, SD, Variance, Co variance, Correlation coefficient, kurtosis		
	2.2	Probability and Probability Distribution: Normal Distribution, Binomial Distribution, Poisson Distribution ,Analysis of Covariance, Data Mining (Association Mining, Apriori with Real data), Cumulative Probability, Continuous probability Distributions, Central Limit Theorem, Baye's Theorem, Naïve Bayes Theorem, Null and alternative Hypotheses, Linear Discriminant Analysis(LDA), QDA.		
3	**Title**	Predictive modeling and Descriptive Modeling	1.2	6
	3.1	Predictive Modeling: Simple Linear Regression, Predictive modeling process, supervised and unsupervised learning, parametric and nonparametric models, business intelligence, challenges in using predictive analytics, Introduction to time series analysis and time series mining, Introduction to spatio-temporal data, spatio-temporal model, fast dynamic time warping.		
	3.2	Descriptive Modeling: Linear Algebra and Matrix computation, Principal components analysis (PCA), singular value decomposition (SVD), probabilistic PCA, applying, PCA to new data, PCA for data interpretation, EM algorithm for PCA, Independent Component Analysis (ICA).		
4	**Title**	Evaluation Methodology for data science	1.2	6
	4.1	Experimental setups, training, tuning, test data, holdout method, cross-validation, bootstrap method. Measuring performance of a model: Accuracy, ROC curves, precision-recall curves, loss functions for regression Interpretation of results: Confidence interval for accuracy, hypothesis tests for comparing models, algorithms.		
5*	**Title**	Self Study		4
	1 2	Text Analytics and Sentiment Analysis Identify where interesting data sets relevant to the following		

		domains and build a suitable data science pipeline show the outcome E.g (a) Books.(b) Horse racing.(c) Stock prices. (d) Risks of diseases.(e) Colleges and universities .(f) Crime rates.(g) Bird watching.		
			Total (* Not Included)	**28**

Laboratory Component (Minimum 10 Laboratory experiments are expected)

Sr. No	Title of the Experiment
1	Setting up python data science Ecosystem and perform comparative study (E.g Anaconda, Google Colab, Kaggle, Jupyter Notebooks, Tensor flow, keras, Theano and MS NLTK, Pytorch, MS Azure)
2	Using Python, explore different types of datasets (CSV and JSON, excel, txt, pdf, xml, multimedia format, medical format).
3	Data preparation process
4	Feature engineering
5	Clustering techniques and its Application
6	Regression techniques and their Application
7	Classification techniques and their Application
8	Case Study is specific domain
9	Mini Project (Phase 1)
10	Mini Project (Phase 2)

Text Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Doing Data Science	First	Cathy O'Neil, Rachel Schutt	O'Reilly	2013
2	Python Data Science Handbook: Essential Tools for Working with Data	First	Jake VanderPlas	O'Reilly	2016

Reference Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	The Data science design Module	ISSN 1868-0941 ISSN 1868-095X (electronic) ISBN 978-3-319-55443-3 ISBN 978-3-319-55444-0 (eBook) DOI 10.1007/978-3-319-55444-0	Steven S,Skien a	Springer	2017

Course (Category) Code	Course Name	Teaching Scheme (Hrs/week)					Credits Assigned			
		L	T	P	O	E	L	T	P	Total
(PE)	Cyber security Technologies	2	0	2	3	7	2	0	1	3
		Examination Scheme								
CS531		Component	ISE		MSE		ESE			Total
		Theory	50		50		100			200
		Laboratory	50		--		50			100

Pre-requisite Course Codes, if any.	Computer network, Linux operating system ,windows operating system
Course Objective: To prepare students for cyber security and digital forensics job profiles in Cyber security Industry, Academia and Research.	
Course Outcomes (CO): ***At the End of the course students will be able to***	
CS531.1	Interpret and classify different cybercrimes.
CS531.2	Analyze and implement cyber security techniques and use best security practices.
CS531.3	Analyze the risk involved in the critical infrastructures.
CS531.4	Apply machine learning techniques in cyber security.
CS531.5	Develop a deeper understanding of existing and future network security problems from a decision and game theoretic perspective

CO-PO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PO1	PO2	PO3
CS531.1			
CS531.2			
CS531.3			
CS531.4			
CS531.5			

CO-PEO/PSO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PEO1	PEO2
CS531.1		
CS531.2		
CS531.3		
CS531.4		
CS531.5		

BLOOM'S Levels Targeted (Pl. Tick appropriate)

Remember	Understand	Apply	Analyze	Evaluate	Create

Theory Component

Module No.	Unit No.	Topics	Ref.	Hrs.
1		**Introduction to cyber security -** Tenets of Cybersecurity-CIA, Cybercrime, classification of cybercrimes, cyber criminals, various cybercrimes - Phishing, DoS/DDoS, Malware, Ransomware, Virus, Website defacement, scanning & sniffing , SQL injection, Buffer overflow, Session Hijacking ,evil twin, wardriving, bluesnarfing, bluebugging, insecure cloud API, Data Breaches in cloud, Abuse of Cloud Services, cyber terrisisum ,etc. Tools and methods used in cybercrimes, Anatomy of hack, ethical hack and hacking phase. **Practical Exercises -** ● Network reconnaissance with the help of open source tools. ● Network scanning with NMAP and Fping. ● Network sniffing using TCPdump, Wireshark and TShark. ● Web Server scanning using Nikto. ● SQL injection, XSS attack. ● DoS/Dos attack using Python Scapy ● Wireless sniffing using Kismet	1,2	6
2		**Cyber security technologies**- Introduction to critical infrastructure and protection. Cryptosystems, PKI and steganography. Identity and Access Management (IAM), Biometric security, multi factor authentication. Intrusion detection and prevention -IDS, IPS, firewall, SIEM, procurement of cyber SOC, IPsec VPN and SSL VPN, Honeypots, User and Entity Behaviour Analytics (UEBA), SOAR. Centralized log management system. **Practical Exercises -** ● iptables configuration to demonstrate firewall. ● Configuration of IDS system - NIDS using snort, HIDS using	1,2, 4,5	6

		ossec and logwatch. ● Building SIEM using Prelude-SIEM, Prelude-correlator, Prewikka. ● Centralized log management using Syslog-ng and Rsyslog ● Implementation of IPsec and VPN using open source. ● Implementing Honeypot ● Wireless IDS Kismet and Net Stumbler ● Advanced persistent threat (APT) analysis using ELK stack. ● Configuring cloud VAPT. ● Implementing PKI using Openssl and pycrypto.		
3		**Risk assessment and management** Defining security risk, security risk= vulnerability x assets x threats. Vulnerability assessment (VA), penetration testing (PT),network VAPT, web VAPT, cloud VAPT, IT infrastructure and inventory management ,threats analysis ,risk residue , risk appetite ,computation of risk matrix, use cases. **Practical Exercises -** ● Vulnerability assessment using nessus, OpenVas, nikto and nmap. ● Penetration testing using metasploit framework. ● Demonstration of SE Toolkit ● Demonstration of cloud VAPT	1,2, 4,5	6
4		**Cyber security and Machine learning-** Anomaly detection using machine learning techniques, use of data science to catch email fraud and spam detection, Botnet detection, .segregating legitimate and lousy URLs. Malware detection, biometric recognition and software vulnerabilities. Knocking down the CAPTCHAs using artificial intelligence. Machine learning to detect financial fraud. **Practical Exercises -** ● Anomaly detection using machine learning technique using KDD cup 1999 dataset, NSL -KDD, UNB datasets ● Credit Card Fraud Detection using Pycaret Library ● Botnet detection using Machine learning techniques ● Knocking down the CAPTCHAs using A.I. ● Detection of Phishing attack ● Security visualization using open source tools.	3	6
5		**Self-Study:**	6	4*

		Game Theoretic approach to cyber-security-Classification of games, Game Theory Methods for Cyber Security Applications, Deterministic Security Games-Security Game Model, Intrusion Detection Games, Sensitivity Analysis, Security Games for Vehicular Networks, Security Games in Wireless Networks. Stochastic Security Games-Markov Security Games, Stochastic Intrusion Detection Game. **Practical Exercises -** ● Implementing honeypot with the help of game theory.		
			Total (*Not Included)	**28**

Sr. No	Title of the Laboratory Exercises
1	Network reconnaissance with the help of open source tools.
2	Information Security Coding using Python and Scapy
3	Vulnerability Assessment and Penetration Testing (VAPT)
4	Firewall and IDS/IPS
5	Cryptography and PKI
6	Anomaly detection using machine learning techniques
7	Building SIEM
8	APT Detection with ELK
9	Security visualization using open source tools
10	Implementing honeypot with the help of game theory.

Text Books:

Sr. No	Title	Authors	Publisher	Year
1	Information Security: Principles and Practice Hardcover.	Mark Stamp	Wiley-Blackwell	2011
2	Securing the Cloud: Cloud Computer Security Techniques and Tactics.	J. R. Winkler	Syngress	2011

3	Hands-On Machine Learning for Cybersecurity: Safeguard your system by making your machines intelligent using the Python ecosystem	Soma Halder and Sinan Ozdemir	Packt Publishing	2018
4	Security Operations Center: Building, Operating, and Maintaining your SOC	Joseph Muniz, Gary McIntyre, Nadhem AlFardan	Cisco Press	2015
5	Wireless and Mobile Network Security	Hakima Chaouchi and Maryline Laurent–Maknavicius	ISTE Ltd.	2009
6	Network Security: A Decision and Game-Theoretic Approach	Tansu Alpcan and Tamer Basar	Cambridge University Press	2010

Course (Category) Code	Course Name	Teaching Scheme (Hrs/week)					Credits Assigned			
		L	T	P	O	E	L	T	P	Total
(SBC)	Advanced Programming Lab	0	0	4	3	7	0	0	2	2
		Examination Scheme								
		Component	ISE		MSE		ESE			Total
CS504		Theory	--		--		--			--
		Laboratory	100		--		100			200

Pre-requisite Course Codes, if any.	-
Course Objective:	
Course Outcomes (CO):*At the End of the course students will be able to*	
CS504.1	Apply Python fundamentals and native data-types for problem solving
CS504.2	Develop Python programs using object oriented programming.
CS504.3	Utilize regular expressions, multithreading, networking, database python modules for problem solving.
CS504.4	Apply data visualization and manipulation with python libraries for data science.

CO-PO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PO1	PO2	PO3
CS504.1			
CS504.2			
CS504.3			
CS504.4			

CO-PEO/PSO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PEO1	PEO2
CS504.1		
CS504.2		
CS504.3		
CS504.4		

BLOOM'S Levels Targeted (Pl. Tick appropriate)

Remember	Understand	Apply	Analyze	Evaluate	Create

Laboratory Component (Minimum 10 Laboratory experiments are expected)

Sr. No	Title of the Experiment
1	**Python Basics** **Content:** o Primitive types: int, float, variables, operators, expressions, statements, input/output statements o Conditionals & loops: if, while and for loops **Aim –** Write a menu-Driven text Applications to solve five problems as a menu-driven text-based application. It presents the user with a set of choices (that, e.g., (1) sum of input numbers, (2) average of input numbers, (3) mean of input numbers, (4) median of input numbers, (2) mode of input numbers and (X) Quit. The user makes a selection, which is then executed. The program exits when the user chooses the "quit" option. The great advantage of a program like this is that it allows the user to run as many iterations of your solutions without necessarily having to restart the same program over and over again.
2	**Derived Data Types** **Content:** o Built-in derived data types: Lists, Tuples, Dictionaries, Strings, Sets and methods of these data types o Errors and Exception handling: try block **Aim –** Write a menu-driven text application to maintain bank accounts of customers using Lists, Dictionary, Strings and Sets. This application handles error and exception using *try block*. The application allows following operations as python functions: i) adding and delete accounts of customers, ii) deposit and withdraw money to/from accounts, iii) list last 3 transactions on a particular accounts and iv) exit from the application.
3	**Functions and Generators** **Content:** o Function Basics : def statement, definition and call, local variables o Function Scope: the LEGB rule, global statement, nested functions, o Function Arguments : passing arguments, special arguments, o Advanced Function Topics: Recursive Functions, Function objects, anonymous functions o Comprehensions and Generations : Generator functions and expressions **Aim –** Write a Merge-Sort algorithm using the concept of Divide-and-Conquers to arrange the elements in ascending orders. Use all the operations of divide and conquers as functions.
4	**Python Modules and Packages** **Content:** o Modules: Python Program Architecture, Module Search Path

	○ Module Coding Basics : Module Creation, Module Usage, Module Namespaces ○ Module Packages: Package Import Basics, Package Relative Imports ○ Advanced Module Topics : Module Design Concepts, Data Hiding in Modules, Mixed Usage Modes - **Aim -** Write a python program that counts the lines and characters in a file (similar in spirit to part of what wc does on Unix). With your text editor, code a Python module called mymod.py that exports three top-level names: a) A *count Lines (name)* function that reads an input file and counts the number of lines in it (hint: *file.readlines* does most of the work for you, and *len* does the rest, though you could count with for and file iterators to support massive files too). b) A *count Chars (name)* function that reads an input file and counts the number of characters in it (hint: *file.read* returns a single string, which may be used in similar ways). c) A *test(name)* function that calls both counting functions with a given input filename. Such a filename generally might be passed in, hardcoded, input with the *input* built-in function.
5	**OOP Concepts** **Content:** ○ Classes and OOP: classes, objects, attributes and methods, defining classes and creating objects ○ Other OOP concepts: inheritance, polymorphism, operator overloading, abstract classes **Aim -** Write a python program to define a class, *Point*, that holds an (x, y) coordinate. The class is in file Shape.py, and now add the following operations the class where p, q, r are Point class and n is any number: i) p = q + r, ii) p += q , ii) p = q - r, iii) p -= q, iv) p = q * n, v) p *= n, vi) p = q / n , vii) p /= n
6	**Python Modules for Data Science** **Content:** ○ Numpy: introduction, numpy arrays, numpy index selection, operations on numpy arrays ○ Scipy: introduction, linalg, stats packages ○ Data Visualization: Matplotlib basics, plotting, different types of plots ○ Pandas: Series, dataframes **Aim –** Write a python program to create csv file/dataset using any five *numpy.random* distribution functions. Apply pandas series, dataframes and methods for processing any csv files/datasets and perform the given operations on the dataset using *numpy*. Further, plot the output of all the given operations using data visualization package.
7	**Regular Expressions** **Content:** ○ Special Symbols and Characters

	○ Regexes and Python **Aim –** Write a python program to recognize the simple calculator operations and then execute the same. The operations include addition, subtractions, multiplication, division, squaring number, square-root of number, and reciprocal of number. It also allows mathematical expressions of any of the above operations mentioned.
8	**Python Multithreading** **Content:** ○ Threads Vs Processes ○ Threads and Python: Global Interpreter Lock, Exiting Threads, Accessing Threads from Python, Python Threading Modules ○ The thread Module ○ The threading Module **Aim –** Write a python program with a function that obtains a byte value and a filename (as parameters or user input) and displays the number of times that byte appears in the file. Suppose now that the input file is extremely large. Multiple readers in a file is acceptable, so modify your solution to create multiple threads that count in different parts of the file such that each thread is responsible for a certain part of the file. Collate the data from each thread and provide the correct total. Use the *timeit* module to time Both the single-threaded and new multithreaded solutions and say something about the difference in performance, if any.
9	**Python networking** **Content:** ○ Client/Server Architecture ○ Sockets: Communication Endpoints ○ Socket Server Module ○ Twisted Framework **Aim –** Write two python programs: Simple half-duplex and Full-Duplex Chat programs. In half-duplex, a connection is made and the service starts, only one person can type. The other participant must wait to get a message before being prompted to enter a message. Once a message is sent, the sender must wait for a reply before being allowed to send another message. One participant will be on the server side; the other will be on the client side. In Full-Duplex, both parties can send and receive independent of each other.
10	**Python database programming** **Content:** ○ Python DB-API: Module Attributes, Connection Objects, Cursor Objects, Type Objects and Constructors, Databases and Python Adapters ○ Python and ORMs ○ Non-Relational Databases: NoSQL, MongoDB, PyMongo **Aim –** Develop an python application for maintaining persons records and chatting

	history of Experiment No. 9. Use a relational and Non-Relational database as the back-end and provide a text-based menu-driven interface to display all information of databases.

Text Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Learning Python	Fifth	Mark Lutz	O'Reilly Media	2013
2	Python for Data Analysis: Data Wrangling with Pandas, Numpy, and Ipython	Second	Wes McKinney	O'Reilly Media	2017
3	Core Python Applications Programming	Third	Wesley J Chun	Prentice Hall	2012

Reference Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Core Python Programming	Second	Wesley J Chun	Prentice Hall	2007
2	Programming in Python 3: A Complete Introduction to the Python Language	Second	Mark Summerfield	Addison-Wesley	2018
3	Introduction to Machine Learning with Python: A Guide for Data Scientists	First	Andreas C. Müller, Sarah Guido	O'Reilly Media, Inc.	2015

Course (Category) Code	Course Name	Teaching Scheme (Hrs/week)					Credits Assigned			
		L	T	P	O	E	L	T	P	Total
(SBC)	Big Data Lab	1	0	2	4	7	1	0	1	2
		Examination Scheme								
		Component		ISE		MSE		ESE		Total
CS505		Theory		--		--		--		--
		Laboratory		100		--		100		200

Pre-requisite Course Codes, if any.	
Course Objective:	
Course Outcomes (CO): *At the End of the course students will be able to*	
CS505.1	Prepare for data summarization, query, and analysis.
CS505.2	Apply data modelling techniques to large data sets.
CS505.3	Create applications for Big Data analytics.
CS505.4	Build a complete business data analytic solution.

CO-PO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PO1	PO2	PO3
CS505.1			
CS505.2			
CS505.3			
CS505.4			

CO-PEO/PSO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PEO1	PEO2
CS505.1		
CS505.2		
CS505.3		
CS505.4		

BLOOM'S Levels Targeted (Pl. Tick appropriate)

Remember	Understand	Apply	Analyze	Evaluate	Create

Theory Component

Module No.	Unit No.	Topics	Ref.	Hrs.
1	**Title**	**Introduction To Big Data and Hadoop**	1	3
	1.1	Types of Digital Data, Introduction to Big Data, Big Data Analytics,		
	1.2	Hadoop Echo System, Hadoop 2.x core components, Analysing Data with Hadoop, Hadoop Streaming, Design of HDFS, Command line Interface, Hadoop file system interface		
2	**Title**	**Map Reduce Framework**	1	2
	2.1	Map Reduce Job Run, failures, Job Scheduling, Shuffle and sort,		
	2.2	Task Execution, Map Reduce Types, format and features		
3	**Title**	**Hadoop Eco systems- Pig**	1,2	3
	3.1	Execution Modes of Pig, Comparison of Pig with Databases, Grunt, User Defined Functions, Data Processing operators, Pig Data Types		
	3.2	Shell and Utility Commands Pig Latin : Relational Operators, File Loaders, Group Operator, COGROUP Operator, Joins and COGROUP Union, Diagnostic Operators, Specialized joins in Pig , Built In Functions - Eval Function, Load and Store Functions, Math function, String Function, Date Function		
4	**Title**	**Hbase and Hive**	2	3
	4.1	NoSQL Databases and HBase ,HBasics, Concepts, Clients, Example		
	4.2	Hbase Versus RDBMS, HBase Data Model, HBase Shell, HBase Client API		
5	**Title**	**Spark Framework and Scala**	9,10	3
	5.1	Components of spark, Resilient distributed databases (RDD), Spark core programming concepts, compilation and execution of spark program.		
	5.2	Introduction to Scala, Basic ProgrammingConstructs of Scala		
6	**Self Study**	Security of Hadoop, Hadoop Security Architecture, Performance Evaluation in HDFS, Map reduce in Cloud Computing, Data Analysis using Pig, Difference in Pig and Hive, Hadoop Mapreduce Vs Apache Spark, Mobile big data analysis using Apache Spark, Hive Shell, Hive Services, Hive Metastore, Comparison with Traditional Databases, HiveQL, Tables, Querying Data and User Defined Functions	1,2	6*
			Total (* Not Included)	**14**

Laboratory Component (Minimum 10 Laboratory experiments are expected)

This Course is based on Project based learning

Sr. No	Title of the Experiment
1	To draw and clarify Hadoop Architecture and Ecosystem with the help of a case study using WorkCount example. To define and install Hadoop. Install the Hadoop and the implementation of Map Reduce programs in Java.
2	To implement the following file management tasks in Hadoop System (HDFS): Adding files and directories, Retrieving files, Deleting files
3	To run a basic Word Count Map Reduce program to understand Map Reduce Paradigm: To count words in a given file, To view the output file, and To calculate execution time.
4	To perform NoSQL database using HBase to create, update and insert.
5	Stop word elimination problem: Input: A large textual file containing one sentences per line. A small file containing the set of stop word (One stop word per Line) Output: A textual file containing the same sentences of the large input file without the words appearing in the small file.
6	Map Reduce : Using movie lens data 1. List all the movies and the number of ratings 2. List all the users and the number of ratings they have done for a movie 3. List all the Movie IDs which have been rated (Movie Id with at least one user rating it) 4. List all the Users who have rated the movies (Users who have rated at least one movie) 5. List of all the User with the max, min, average ratings they have given against any movie 6. List all the Movies with the max, min, average ratings given by any user
7	Install and Run Pig then write Pig Latin scripts to sort, group, join, project, and filter your data.
8	Write a Pig Latin scripts for finding TF-IDF value for book dataset (A corpus of eBooks available at: Project Gutenberg)
9	Install and Run Hive then use Hive to create, alter, and drop databases, tables, views, functions, and indexes.
10	Extract facts using Hive OR Extract sessions using Pig Hive allows for the manipulation of data in HDFS using a variant of SQL. This makes it excellent for transforming and consolidating data for load into a relational database. In this exercise you will use HiveQL to filter and aggregate click data to build facts about user's movie preferences. The query results will be saved in a staging table used to populate the Oracle Database.
11	Write a Map Reduce program that mines weather data. Weather sensors collecting data every hour at many locations across the globe gather large volume of log data, which is a good candidate for analysis with MapReduce, since it is semi structured and record-oriented. Data available at: https://github.com/tomwhite/hadoopbook/tree/master/input/ncdc/all.

	---Find average, max and min temperature for each year in NCDC data set? ---Filter the readings of a set based on value of the measurement, Output the line of input files associated with a temperature value greater than 30.0 and store it in a separate file.
12	Purchases.txt Dataset: --Instead of breaking the sales down by store, give us a sales breakdown by product category across all of our stores. What is the value of total sales for the following categories? --Toys --Consumer Electronics Find the monetary value for the highest individual sale for each separate store. What are the values for the following stores? Reno, Toledo, Chandler Find the total sales value across all the stores, and the total number of sales.
13	Install, Deploy & configure Apache Spark Cluster. Run apache spark applications using Scala.
14	Data analytics using Apache Spark on Amazon food dataset, find all the pairs of items frequently reviewed together. Write a single Spark application that: --Transposes the original Amazon food dataset, obtaining a PairRDD of the type: <user_id>→ <list of the product_ids reviewed by user_id> --Counts the frequencies of all the pairs of products reviewed together; --Writes on the output folder all the pairs of products that appear more than once and their frequencies. The pairs of products must be sorted by frequency.

Text Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Big Data, Black Book (covers Hadoop 2, MapReduce, Hive, Yarn, Pig, R And Data Visualization, Black Book, Dreamtech	First	Dreamtech	Dreamteck	2016
2	Big Data and Analytics, Paperback 2019	Second	Subhashini Chellappan Seema Acharya	Wiley	2019

Reference Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Hadoop: The Definitive Guide	Third	Tom White,	O' Reilley	2012
2	Hadoop Operations	Second	Eric Sammer	O'Reilley	2012

3	Big data analytics with R and Hadoop	Second	Vignesh Prajapati	SPD	2013
4	Programming Hive	Second	E. Capriolo, D. Wampler, and J. Rutherglen	O'Reilley	2012
5	Programming Pig	Second	Alan Gates	O'Reilley	2011
6	HBase: The Definitive Guide	Third	Lars George	O'Reilley	2011
7	Programming in Scala	Second	Martin Odersky, Lex Spoon, Bill Venners	artima	2016
8	Learning Spark	Second	Matei Zaharia, Patrick Wendell, Andy Konwinski, Holden Karau	O'Reilley	2015

Semester-II

Course (Category) Code	Course Name	Teaching Scheme (Hrs/week)					Credits Assigned			
		L	T	P	O	E	L	T	P	Total
(PC)	Advanced Network and Communication System	3	0	0	5	8	3	0	0	3
CS506		Examination Scheme								
		Component		ISE		MSE		ESE		Total
		Theory		75		75		150		300
		Laboratory		--		--		--		--

Pre-requisite Course Codes, if any.	Computer Network
Course Objective:	
Course Outcomes (CO):*At the End of the course students will be able to*	
CS506.1	Understand the issues in the protocol design and apply it to Quality of service in communication System.
CS506.2	Understand and Apply the concept of Routing and IPV6 for Communication and Design.
CS506.3	Identify challenges in designing various protocols and analyze different protocols for Wireless Ad-hoc Networks
CS506.4	Understand the basics of software defined networking and explore research problems in that area.
CS506.5	Determine the communication pattern and network technology for High Performance Computing for communication and Design.

CO-PO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PO1	PO2	PO3
CS506.1			
CS506.2			
CS506.3			
CS506.4			
CS506.5			

CO-PEO/PSO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PEO1	PEO2
CS506.1		
CS506.2		
CS506.3		
CS506.4		
CS506.5		

BLOOM'S Levels Targeted (Pl. Tick appropriate)

Remember	Understand	Apply	Analyze	Evaluate	Create

Theory Component

Module No.	Unit No.	Topics	Ref.	Hrs.
1	Title	**Internetworking**	1,3	08
	1.1	Congestion control and Resource allocation: Issues of Resource Allocation, Queuing Disciplines: FIFO, Fair Queuing, TCP Congestion Control: Additive Increase/Multiplicative Decrease, Slow Start, Fast Retransmit and Fast Recovery.		
	1.2	Congestion-Avoidance Mechanisms: DECbit, Random Early Detection (RED), Source-Based Congestion Avoidance, Quality of Service: Application Requirements, Integrated Services (RSVP), Differentiated Services (EF, AF)		
2	Title	**Routing and IPV6**	3,4	08
	2.1	IPv4 Routing Principles, Routing Information Protocol (RIP), IGRP and EIGRP, OSPF for IPv4 and IPv6, Border Gateway Protocol (BGP)		
	2.2	Pv4 deficiencies, IPv6 addressing, multicast, Anycast, ICMPv6, Neighbour Discovery, Routing, Resource Reservation, IPv6 protocols		
3	Title	**Ad Hoc Wireless Networks**	2,5	09
	3.1	**Ad-hoc Wireless Media Access Protocols**: Problems in Ad-hoc Channel Access, Receiver- Initiated MAC Protocols, Sender-Initiated MAC Protocols, Ad-hoc MAC Protocols- Multiple Access with Collision Avoidance (MACA), MACA-BI (By Invitation), Power-Aware Multi-Access Protocol with Signaling (PAMAS), Dual Busy Tone Multiple Access (DBTMA), MARCH: Media Access with Reduced Handshake.		
	3.2	**Routing Protocols for Ad-hoc Networks:** Table Driven Routing Protocols- Destination Sequenced Distance Vector (DSDV), Source-Initiated On-Demand Approaches- Ad-hoc On- Demand Distance Vector Routing (AODV), Dynamic Source Routing (DSR), Hybrid Routing Protocols- Zone Routing Protocol (ZRP)		
4	Title	**Ad Hoc Wireless Networks Cont.**	2,3.,5	05
	4.1	**TCP over Ad-hoc Wireless Networks:** TCP Flow Control, TCP Congestion Control, Issues with TCP. Versions of TCP, Problems facing TCP in Wireless Ad-hoc,		
	4.2	Approaches to TCP over Ad-hoc - TCP Feedback (TCP-F), TCP with Explicit Link Failure Notification, TCP-BuS		

5	**Title**	**Performance and Communication Operations**		
	5.1	Parallel Architecture, Architecture Classification Schemes, Performance, Performance Metrics, Parallel Programming Model		
	5.2	Operations - One-to-All Broadcast and All-to-One Reduction, All-to-All Broadcast and Reduction, All-Reduce and Prefix-Sum Operations, Scatter and Gather, All-to-All Personalized Communication, Circular Shift, Improving the Speed of Some Communication Operations	6,7	**12**
6*	**Self Study**	Interconnection Network: Permutation and used, Network Classification, Non-blocking, Parallel Architecture and cognitive Function: ANM, NN Classifier, STPN, SLT model. Software Defined Networking: Introduction to Software Defined Networking, Control and Data Planes, SDN Controllers. Introduction to Open flow Protocol, Network Function Virtualization-Concepts	6,9	**6**
			Total (* Not Included)	**42**

Text Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Computer Networks: A Systems Approach	Forth	Larry L. Peterson and Bruce S. Davie	Elsevier	2007
2	Ad Hoc Wireless Networks: Architectures and Protocols	First	C. Siva Ram Murthy, B.S. Manoj	Pearson	2004

Reference Books

Sr. No	Title	Edition	Authors	Publisher	Year
3	TCP / IP: The Ultimate Protocol Guide Applications, Access and Data Security – Vol .2	Second	Philip M. Miller	Wiley	2009
4	IPv6: Theory, Protocols and Practice, Morgan Kaufmann, 2nd Edition, 2004.	Second	Pete Loshin	Morgan Kaufmann	2004
5	Ad hoc Mobile Wireless Networks – Protocols and Systems	First	C. K. Toh	Pearson	2007
6	Parallel Computing	Second	Moreshwar Bhujade	New Age Science	2009
7	Introduction to Parallel Computing	Second	Ananth Grama	Addison Wesley	2003
8	Advanced Computer Architecture: Parallelism, Scalability, Programmability	Second	Kai Hwang, Naresh Jotwani	Mcgraw-Hill Education	2008
9	Software Defined Networking	First	Thomas D NAdeau and Ken Grey	O'Reilly	2013

Course (Category) Code	Course Name	Teaching Scheme (Hrs/week)					Credits Assigned			
		L	T	P	O	E	L	T	P	Total
(PC)	Distributed Computing Systems	3	0	0	5	8	3	-	-	3
CS507		Examination Scheme								
		Component		ISE		MSE		ESE		Total
		Theory		75		75		150		300
		Laboratory		--		--		--		--

Pre-requisite Course Codes, if any.	Operating Systems, Computer Networks
Course Objective:	
Course Outcomes (CO):*At the End of the course students will be able to*	
CS507.1	Describe the problems and challenges associated with the principles of distributed computing.
CS507.2	Apply Distributed Computing techniques, Synchronous and Processes.
CS507.3	Analyze different algorithm for mutual exclusion and deadlock detection
CS507.4	Illustrate the peer to peer computing concepts
CS507.5	Illustrate the concept of blockchain

CO-PO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PO1	PO2	PO3
CS507.1			
CS507.2			
CS507.3			
CS507.4			
CS507.5			

CO-PEO/PSO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PEO1	PEO2
CS507.1		
CS507.2		
CS507.3		
CS507.4		
CS507.5		

BLOOM'S Levels Targeted (Pl. Tick appropriate)

Remember	Understand	Apply	Analyze	Evaluate	Create

Theory Component

Module No.	Unit No.	Topics	Ref.	Hrs.
1	Title	**Introduction**	1,2	6
	1.1	Definition , Motivation, message passing system versus shared memory systems, Primitives for distributed communication, Synchronous versus asynchronous executions, Design issues and challenges, examples of Distributed systems.		
	1.2	A distributed program, A model of distributed executions, Models of communication networks, Global state of a distributed system, Cuts of a distributed computation, Models of process communications.		
2	Title	**Clock Synchronization and message Ordering**		8
	2.1	Physical clock, Physical clock synchronization, Sequential and Concurrent events , Logical clocks, vector clocks, Logical clock synchronization, Election algorithms.	2	
	2.2	Message ordering paradigms , Asynchronous execution with synchronous communication, Synchronous program order on an asynchronous system , Group communication Causal order (CO), Total order, Termination detection using distributed snapshots, Message-optimal termination detection, Termination detection in a very general distributed computing model.	1	
3	Title	**Distributed mutual exclusion algorithms and Deadlock Detection**	1	10
	3.1	Lamport's algorithm , Ricart–Agrawala algorithm, Singhal's dynamic information-structure algorithm, Lodha and Kshemkalyani's fair mutual exclusion algorithm , Quorum-based mutual exclusion algorithms , Maekawa's algorithm , Agarwal–El Abbadi quorum-based algorithm, Token-based algorithms, Suzuki–Kasami's broadcast algorithm		
	3.2	Models of deadlocks, Knapp's classification of distributed deadlock detection algorithms, Mitchell and Merritt's algorithm for the single resource model, Chandy–Misra–Haas algorithm for the AND model, Chandy–Misra–Haas algorithm for the OR model, Kshemkalyani–Singhal algorithm for the P-out-of-Q model		
4	Title	**Peer-to-peer computing and overlay graphs**	1	8
	4.1	Introduction, Data indexing and overlays, Unstructured overlays, Chord distributed hash table, Content addressable networks, challenges in P2P system design.		
	4.2	Tradeoffs between table storage and route lengths, Graph structures of complex networks, Internet graphs, Generalized random graph networks, Small-world networks.		

5	**Title**	**Blockchain**	3	
	5.1	The history of blockchain and Bitcoin, Blockchain defined, Generic elements of a blockchain, How blockchain works, Benefits and Limitations of blockchain.		**10**
	5.2	Types of blockchain, Decentralization using blockchain, Methods of decentralization, Routes to decentralization, Decentralization framework example.		
6	**Self Study**	Distributed shared memory, Fault-Tolerant Systems, Distributed Discrete-Event Simulation.	2	**6***
			Total (* Not Included)	**42**

Text Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Distributed Computing Principles, Algorithms, and Systems	First	Ajay D. Kshemkalyani, Mukesh Singhal	Cambridge University Press	2011
2	Distributed Systems An Algorithmic Approach	Second	Sukumar Ghosh	CRC Press	2015
3	Mastering Blockchain	Second	Imran Bashir	Packt Publishing Ltd	2018

Reference Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Distributed Systems: Principles and Paradigms	Second	A S Tanenbaum, Martin Steen	PHI	2006

Course (Category) Code	Course Name	Teaching Scheme (Hrs/week)					Credits Assigned			
		L	T	P	O	E	L	T	P	Total
(PE)	Machine Learning	2	0	2	3	7	2	0	1	3
		Examination Scheme								
		Component	ISE		MSE		ESE			Total
CS512		Theory	50		50		100			200
		Laboratory	50		--		50			100

Pre-requisite Course Codes, if any.	
Course Objective: To design and analyze various machine learning algorithms and techniques with a modern outlook focusing on recent advances. Explore supervised and unsupervised learning paradigms of machine learning.	
Course Outcomes (CO):*At the End of the course students will be able to*	
CS512.1	Exhibit the knowledge about machine learning basics
CS512.2	Illustrate various supervised and unsupervised machine learning methods such as regression, trees, classification and clustering
CS512.3	Evaluate regression, classification techniques for a given problem
CS512.4	Evaluate unsupervised learning techniques for a given problem.

CO-PO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PO1	PO2	PO3
CS512.1			
CS512.2			
CS512.3			
CS512.4			

CO-PEO/PSO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PEO1	PEO2
CS512.1		
CS512.2		
CS512.3		
CS512.4		

BLOOM'S Levels Targeted (Pl. Tick appropriate)

Remember	Understand	Apply	Analyze	Evaluate	Create

Theory Component

Module No.	Unit No.	Topics	Ref.	Hrs.
1	**Title**	**Machine learning basics**	1,2,4,7	**04**
	1.1	What is machine learning? Key terminology, Key tasks of machine learning, How to choose the right algorithm, Steps in developing a machine learning application.		
2	**Title**	**Learning with regression and trees**	1,2,6,7	**06**
	2.1	Predicting numeric values: linear regression		
	2.2	Learning with Trees: Decision Trees, Constructing Decision Trees using Gini Index, Classification and Regression Trees (CART).		
3	**Title**	**Learning with Classification**	1,2,6,7	**10**
	3.1	Classification with logistic regression with sigmoid function, Nearest neighbours and Naïve Bayes, Measuring classifier accuracy.		
	3.2	Support vector machines: Introduction to SVM, Hyperplane, vectors.		
4	**Title**	**Unsupervised Learning**	1,2,7	**08**
	4.1	Clustering, k-means algorithm, k-medoids, Principal Component Analysis, Single value decomposition.		
5*	**Self Study**	Evaluating Machine Learning algorithms and Model Selection, Recent trends in various learning techniques of machine learning and classification methods for various applications, Introduction to Statistical Learning Theory, Ensemble Methods (Boosting, Bagging, and Random Forests).		
			Total (* Not Included)	**28**

Laboratory Component (Minimum 10 Laboratory experiments are expected)

Sr. No	Title of the Experiment
1	Implement Linear regression
2	Implement any two Supervised Algorithm
3	Implement any one Unsupervised Algorithm
4	Comparison of Machine Learning Algorithm

5	Based on the syllabus topics and any other recent topic from Machine Learning area. Evaluation of Mini-Project will be based on the following rubrics: Performance Indicator Identification of problem Implementation Result interpretation Presentation and Report Writing

Text Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Machine Learning In Action	-	Peter Harrington	Manning Publications	2012
2	Machine Learning	First	Tom M. Mitchell	McGraw Hill	1997
3	Machine Learning: A Probabilistic Perspective	First	Kevin Murphy	MIT Press	2012

Reference Books

Sr. No	Title	Edition	Authors	Publisher	Year
4	Pattern Recognition and Machine Learning	-	Christopher Bishop	Springer	2007
5	The Elements of Statistical Learning, Data Mining, Inference, and Prediction.	Second	Trevor Hastie, Robert Tibshirani, Jerome Friedman	Springer	2009
6	Machine Learning An Algorithmic Perspective	Second	Stephen Marsland	CRC Press	2014
7	Introduction to Machine Learning	-	Ethem Alpaydın	MIT Press	

Course (Category) Code	Course Name	Teaching Scheme (Hrs/week)					Credits Assigned			
		L	T	P	O	E	L	T	P	Total
(PC)	Machine Intelligence	2	0	2	3	7	2	0	1	3
CS522		Examination Scheme								
		Component		ISE		MSE		ESE		Total
		Theory		50		50		100		200
		Laboratory		50		--		50		100

Pre-requisite Course Codes, if any.	
Course Objective: To apply skills of machine learning algorithms on real-life applications	
Course Outcomes (CO):*At the End of the course students will be able to*	
CS522.1	To select appropriate Data Visualization techniques for a given scenario.
CS522.2	Evaluate the performance of Classification/ Clustering algorithms.
CS522.3	Apply Artificial Neural Networks on large dataset.
CS522.4	Implement Forecasting algorithms to solve real-world problems.

CO-PO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PO1	PO2	PO3
CS522.1			
CS522.2			
CS522.3			
CS522.4			

CO-PEO/PSO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PEO1	PEO2
CS522.1		
CS522.2		
CS522.3		
CS522.4		

BLOOM'S Levels Targeted (Pl. Tick appropriate)

Remember	Understand	Apply	Analyze	Evaluate	Create

Theory

Module No.	Unit No.	Topics	Ref.	Hrs.
1		**Visualization**	1	2
	1.1	Introduction to data visualization		
	1.2	When, Why and Where to use: Bar Graph, Pie chart, Dot plot, Line graph, Scatter plot, Histogram (uniform, non-uniform width), Box Plot, Heat Map		
2		**Supervised Machine Learning**	2	6
	2.1	Introduction to Machine Learning, Types- Supervised, Un supervised, Semi-supervised		
	2.2	Basic Concepts, Classification methods: Decision Tree Induction, Attribute Selection Measures, Tree pruning. Bayesian Classification: Naïve Baye's Classifier, Support Vector Machine, Bayesian Belief Networks, Hidden Markov Model		
	2.3	Ensemble Methods: Bagging, Boosting- AdaBoost, , Random Forests		
3		**Unsupervised Machine Learning**	2	5
	3.1	Cluster Analysis: Basic Concepts, Partitioning Methods: K-Means, K-Medoids. Hierarchical Methods: Agglomerative, Divisive, BIRCH		
	3.2	Density-Based Methods: DBSCAN, OPTICS		
	3.3	Outlier Detection Methods: Supervised, Semi-Supervised, Unsupervised, Proximity based, Clustering Based		
4		**Artificial Neural Networks**	3	8
	4.1	Fundamental Concepts and Models of Artificial Neural Systems: Biological Neurons and Their Artificial Models, Models of Artificial Neural Networks, Neural Processing, Learning and Adaptation, Neural Network, Learning Rules and Comparison.		
	4.2	Linearly and Non-Linearly Separable Pattern Classification.		
	4.3	Perceptron Convergence Theorem		
	4.4	Multi-layer Feed forward Network: Delta Learning Rule for Multi-Perceptron Layer, Generalized Delta Learning Rule, Feed forward, Recall and Error, Back-propagation Training, Learning Factors		
5		**Forecasting**	4	7
	5.1	Basics of Time Series, Defining the forecasting problem , Holt-Winters' seasonal method		
	5.2	Correlation analysis, simple linear regression, statistical tests of significance, multiple regression, causal factors in multiple regression, statistical characteristics of this method.		
	5.3	Short-range forecasting, Analysis of Time Series data-plots, seasonality, Auto correlation, ARIMA Models		

6*	**Self Study**	Case Studies: Pinterest – Improved Content Discovery, Twitter – Curated Timelines, HubSpot – Smarter Sales, Google, Salesforce – Intelligent CRMs, LinkedIn Deep Learning: Introduction, Convolutional Neural Network, Recurrent Neural Network, Generative Adversarial Network, Reinforcement learning (Markov Decision Processes, Value Iteration)Q-learning	R1	**5***
Total				**28**

*** Total 28 hrs. does not include this module's hours.**

Laboratory:

Sr. No	Title of the Experiment
1	Data Visualization: Use appropriate data set and demonstrate various charts
2	Supervised Learning: Choose appropriate dataset for internet traffic analysis and apply algorithms like: naïve Bayes, Support Vector Machine. Analyze and compare their performance metrics.
3	Supervised Learning: For credit card fraud detection, choose a dataset and apply Decision tree, Random Forest algorithm. Measure and evaluate model performance.
4	Unsupervised learning: Choose appropriate dataset for document classification and form appropriate clusters using partition based clustering algorithms and evaluate performance metrics.
5	Unsupervised learning: Form appropriate clusters using hierarchical clustering algorithm and density based clustering, analyze and compare their performance metrics.
6	Neural Networks: Apply neural networks for character recognition.
7	Neural Networks: Apply Multilayer Perceptron for a scenario like Autonomous Driving - Image Recognition, Object detection, Route Adjustment, and evaluate model performance.
8	Neural Networks: Apply back propagation algorithm for face recognition and evaluate model performance.
9	Time Series: Perform analysis on Covid-19 dataset with the help of Holt and Winter Model.
10	Time Series: Analyze Share Market with ARIMA Model and forecast potential good stocks.

Text Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Storytelling with data	First	Cole Nussbaumer Knaflic	Wiley	2015
2	Introduction to Machine Learning with Python: A Guide for Data Scientists	First	Müller, Andreas C., Guido, Sarah	O'Reilly	2016
3	Introduction to Artificial Neural Systems	First	Jacek M. Zurada	Jaico Publishing House	2000
4	Time series analysis and its applications	Fifth	Robert H. Shumway	Springer	2000

Reference Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Deep Learning	First	Ian Goodfellow, Yoshua Bengio and Aaron Courville	The MIT Press	2017

Course (Category) Code	Course Name	Teaching Scheme (Hrs/week)					Credits Assigned			
		L	T	P	O	E	L	T	P	Total
PE	Digital Forensics and Incident Response	2	0	2	3	7	2	0	1	3
		Examination Scheme								
		Component	ISE		MSE		ESE			Total
CS532		Theory	50		50		100			200
		Laboratory	50		--		50			100

Pre-requisite Course Codes, if any.	Cyber Security and Digital Forensics
Course Objective: To prepare students for cyber security and digital forensics job profiles in Cyber security Industry, Academia and Research.	
Course Outcomes (CO): ***At the End of the course students will be able to***	
CS532.1	To analyze the hardware, software, firmware and tools etc for forensic investigation processes.
CS532.2	Create documents and maintain chain of custody.
CS532.3	Develop Python scripting to perform a variety of forensic collection and analysis tasks.
CS532.4	Develop digital forensics is part of the incident response (IR) capability Incident Response is an integral part of information Assurance (IA) and Forensic readiness. Create Evidence-centric procedures and processes.
CS532.5	Apply appropriate domestic and international law.

CO-PO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PO1	PO2	PO3
CS532.1			
CS532.2			
CS532.3			
CS532.4			
CS532.5			

CO-PEO/PSO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PEO1	PEO2
CS532.1		
CS532.2		
CS532.3		
CS532.4		
CS532.5		

BLOOM'S Levels Targeted (Pl. Tick appropriate)

Remember	Understand	Apply	Analyze	Evaluate	Create

Theory Component

Module No.	Unit No.	Topics	Ref.	Hrs.
1		**Introduction to Digital Forensics:** Cyberspace and criminal behaviour, cyber crime investigation, forensics science, forensics methodologies, cyber incident handling and response, Controlling Contamination-the Chain of Custody, Legal perspectives and security compliance of digital forensics. ISO standards relating to Digital Evidence, SWDGE Model SOP for Computer Forensics, NIST Digital Evidence.	1,2	**6**
2		Setting up an electronic evidence forensics laboratory, Phases of Forensics. Grow your own tools for Computer Forensics with help Python Libraries. **Practical Exercises:** To develop Python scripting to perform a variety of forensic collection and analysis tasks. To demonstrate using Python to work with encrypted files, to extract metadata, to examine windows artifacts, to track Web and email usage, to foot print applications, to carve artifacts from volatile memory, to carve file systems, and to analyze network traffic. Implementing Python frameworks for development of further tools.	3,4	**6**
3		**Network Forensics:** Applying Forensic Science to Networks Preparation and Authorization, Identification Documentation, Collection, and Preservation Filtering and Data Reduction, Class/ Individual Characteristics and Evaluation of Source, Evidence Recovery, Investigative Reconstruction, Reporting Results Network Forensics Analysis Toolkit (NFAT) **Practical Exercises:** Collection of network traffic data (Live acquisition and dump) Analyzing the traffic and persevering original. Implementing network sensors. Correlate with host sensors. Develop an algorithm to filter and extract anomalous traffic. Use open source and commercial domain tools for NFA. Event reconstruction. OSINT Tools-tshark, wireshark, tcpdump & xplico **Windows Forensics:** Digital Evidence on Windows Systems, File Systems, Data Recovery, Log Files, Registry, Internet Traces, Program Analysis.	3,4, 5,6	**6**

		Tools and methods. **UNIX/Linux OS Forensics:** UNIX Evidence Acquisition Boot Disk, File Systems, Overview of Digital Evidence Processing Tools, Data Recovery, Log Files, File System Traces Internet Traces Tools and methods. **Practical Exercises:** To perform live analysis, capture volatile data, make images of media, analyze filesystems, analyze network traffic, analyze files, perform memory analysis, and analyze malware for a Windows subject on a Linux system with readily available free and open source tools. Windows Registry analysis. To build an in-house forensic capability via a variety of free, open-source, and commercial tools provided within the SANS Windows SIFT Workstation, NST etc		
4		**Malware Forensics, Memory Forensics and Disk Forensics:** File Identification and Profiling in Windows and Linux Analysis of a Suspect Program, Discovering and Extracting Malware and Associated Artifacts from Linux Systems and Windows systems, Memory Forensics: Analyzing Physical and Process, Memory Dumps for Malware Artifacts, Malware Incident Response: Volatile Data Collection and Examination on a Live Windows Systems and Linux systems. Tools and techniques of conducting runtime behavioral malware analysis and static code analysis. **Practical Exercises:** Collecting data from system memory (system registers, cache, RAM) in raw form and then carving the data from Raw dump. Extracting data from storage media by searching active, modified, or deleted files. OSINT tools, TCT, Autopsy, Sleuth-kit, volatility tools. **Email Forensics:** Broad steps in email forensics, Investigate Email Headers, Sender IP address, Verify Emails, Fake Email Investigation, Hacked, Email Scams. Tools and methods used in Email Forensics. **Practical Exercises:** Identification of malicious code, to study their payload, viruses, worms, etc. Recovery and analysis of emails, including deleted emails, calendars, and contacts	6,7, 8	**6**
5		**Self-Study:**	7,8	**4***

		Anti Forensics Anti-forensics techniques, detection and countermeasures **Incident Handling and Incident Response** Development of Incident Handling and Response methodology (PDCAERF) Incident Response is an integral part of information Assurance (IA) Forensic readiness. Evidence-centric procedures and processes. Proper evidence handling and management, Determining the scope Containment strategies. Event reconstruction, Review the incident. Follow up reporting. **Practical Exercises:** To develop digital forensics is part of the incident response (IR) Capability & Automated report mechanism.		
			Total (*Not Included)	**28**

Sr. No	Title of the Laboratory Exercises
1	Setting up an electronic evidence forensics laboratory
2	Network Forensics using Open Source Tools-NFAT
3	Computer Forensics
4	To develop Python scripting to perform a variety of forensic collection and analysis tasks.
5	Memory Forensics
6	Disk Forensics & Email Forensics
7	Malware Forensics
8	Windows Forensics
9	Linux Forensics
10	To develop digital forensics is part of the incident response (IR) capability & Automated report mechanism.

Text Books:

Sr. No	Title	Authors	Publisher	Year
1	Crime Scene Forensics A Scientific Method Approach	Robert C Shaler	CRC Press	2011
2	Computer Forensics and Cyber Crime: An Introduction	Marjie T. Britz	Pearson	2013
3	Computer Forensics Evidence Collection & Preservation	EC-Council	Cengage Learning	2010
4	Digital Evidence and Computer Crime: Forensic Science, Computers, and the Internet	Eoghan Casey BS MA	Academic Press	2011

5	Computer Forensics: Computer Crime Scene Investigation	John R. Vacca	Charles River Media	2002
6	Malware Forensics: Investigating and Analyzing Malicious Code	Cameron H. Malin, Eoghan Casey, James M. Aquilina	Syngress	2008
7	Introduction to Network Forensics	The European Union Agency for Network and Information Security (ENISA)	ENISA	2019
8	Windows Forensics: The Field Guide for Corporate Computer Investigations	Chad Steel	John Wiley & Sons	2006

Course (Category) Code	Course Name	Teaching Scheme (Hrs/week)					Credits Assigned			
		L	T	P	O	E	L	T	P	Total
(PE)	Deep Learning	2	0	2	3	7	2	0	1	3
CS513		Examination Scheme								
		Component	ISE		MSE		ESE			Total
		Theory	50		50		100			200
		Laboratory	50		--		50			100

Pre-requisite Course Codes, if any.	ACI, Soft Computing , Machine Learning
Course Objective: To explore deep learning techniques with different learning strategies.	
Course Outcomes (CO):*At the End of the course students will be able to*	
CS513.1	Identify the deep learning algorithms which are more appropriate for various types of learning tasks in various domains
CS513.2	Implement CNN using Regularization techniques.
CS513.3	Analyze different deep learning strategies
CS513.4	Create Deep learning Models for real-world problems

CO-PO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PO1	PO2	PO3
CS513.1			
CS513.2			
CS513.3			
CS513.4			

CO-PEO/PSO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PEO1	PEO2
CS513.1		
CS513.2		
CS513.3		
CS513.4		

BLOOM'S Levels Targeted (Pl. Tick appropriate)

Remember	Understand	Apply	Analyze	Evaluate	Create

Theory Component

Module No.	Unit No.	Topics	Ref.	Hrs.
1	**Title**	**Convolutional Neural Network**	2	**6**
	1.1	Foundations of Convolutional Neural Network, The Convolution Operation, Parameter sharing, Feature Map, Pooling		
	1.2	Visualizing Convolutional Neural Network		
	1.3	Regularization techniques like adding a dropout layer, adding laplacian, gaussian noise		
2	**Title**	**Deep Neural Network:**	1, 4	**8**
	2.1	Deep Learning Intuition, Neural Network Basics, Shallow Neural Network		
	2.2	Deep Feed Forward network, Basic L1 L2 regularizations		
	2.3	Optimization algorithms, Deep Reinforcement Learning, deep Q networks		
	2.4	Hyper-parameter Tuning, Batch Normalization		
3	**Title**	**Deep Learning Models**	2, 3	**8**
	3.1	Recurrent neural networks (RNN)		
	3.2	Long short term memory networks (LSTM)		
	3.3	Auto encoders (AE)		
4	**Title**	**Deep learning strategies**	3	**6**
	4.1	Restrictive Boltzmann Machines (RBMs), MCMC and Gibbs Sampling, Sigmoid net.		
	4.2	Deep Boltzmann Machines, Unsupervised Learning		
	4.3	Gradient computations in RBMs		
5*	**Self Study**	LeNet, AlexNet, ResNet Dot product attention, Seq2Seq networks Generative adversarial networks (GAN), Application of GAN Building Deep learning Models for Digit Recognition, Handwritten character recognition, Image classification, Object Tracking Building Deep learning Models - Medical Imaging - Object Tracking - Image Caption Transfer Learning techniques.		
			Total (* Not Included)	**28**

Laboratory Component (Minimum 10 Laboratory experiments are expected)

Sr. No	Title of the Experiment
1	Understanding ANN using Tensor Flow
2	Semantic Segmentation in TensorFlow with Fully Connected Neural Networks
3	Visualizing Convolutional Neural Network using Tensor Flow with Keras Data. Understanding Regularization techniques for the same
4	Object detection using RNN using Tensor Flow

5	A Restricted Boltzmann Implementation in TensorFlow
6	Zero-sum Game using Generative Adversarial Networks
7	Mini Project will be based on syllabus topic and Digit Recognition, Handwritten character recognition, Image classification, Object Tracking etc, and any other topic in recent trends of deep learning. Evaluation of Mini-Project will be based on the following rubrics Performance Indicator Identification of problem Implementation Presentation Report Writing

Text Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Deep Learning	-	Ian Goodfellow Yoshua Bengio, and Aaron Courville	MIT Press	2016
2	Advanced Applied Deep Learning: Convolutional Neural Networks and Object Detection	First	Umberto Michelucci		2019

Reference Books

Sr. No	Title	Edition	Authors	Publisher	Year
3	Generative Deep Learning: Teaching Machines to Paint, Write, Compose, and Play	First	David Foster	O'Relly	
4	Deep Reinforcement Learning Hands-On: Apply modern RL methods, with deep Q- networks, value iteration, policy gradients, TRPO, AlphaGo Zero and more	Kindle Edition	Maxim Lapan	Packt	2018
5	Deep learning	-	Bengio, Yoshua, Ian J. Goodfellow, and Aaron Courville	MIT Press	2015
6	Pro Deep Learning with TensorFlow A Mathematical Approach to Advanced Artificial Intelligence in Python	-	SantanuPattanaya k	APress	-

Course (Category) Code	Course Name	Teaching Scheme (Hrs/week)					Credits Assigned			
		L	T	P	O	E	L	T	P	Total
(PE)	Explainable AI	2	0	2	3	7	2	0	1	3
CS523		Examination Scheme								
		Component	ISE		MSE		ESE			Total
		Theory	50		50		100			200
		Laboratory	50		--		50			100

Pre-requisite Course Codes, if any.	
Course Objective:	
Course Outcomes (CO):*At the End of the course students will be able to*	
CS523.1	Apply basic methods and algorithms from area of explainable artificial intelligence.
CS523.2	Demonstrate ideas behind explainable AI and its usage.
CS523.3	Formulate problems as problems from area of artificial intelligence or improve existing code using learned methods.
CS523.4	Evaluate applications and background algorithms used for their implementation through used cases.

CO-PO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PO1	PO2	PO3
CS523.1			
CS523.2			
CS523.3			
CS523.4			

CO-PEO/PSO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PEO1	PEO2
CS523.1		
CS523.2		
CS523.3		
CS523.4		

BLOOM'S Levels Targeted (Pl. Tick appropriate)

Remember	Understand	Apply	Analyze	Evaluate	Create

Theory Component

Module No.	Unit No.	Topics	Ref.	Hrs.
1	Title	**Method for interpreting AI systems**:	1	6
	1.1	Neural Networks via feature visualization, Interpretable Text-Image synthesis, Unsupervised Discrete Representation Learning, Towards Reverse-Engineering Black-Box Neural Networks.		
2	Title	**Explaining the Decisions of AI Systems**:	1	6
	2.1	Explanations for Attributing Deep Neural Network Predictions, Gradient-Based Attribution Methods, Layer-Wise Relevance Propagation: An Overview, Explaining and Interpreting LSTMs		
3	Title	**Evaluating Interpretability and Explanations**:	1	6
	3.1	Comparing the Interpretability of Deep Networks via Network Dissection, Gradient-Based Vs. Propagation-Based Explanations: An Axiomatic Comparison, The (Un)reliability of Saliency Methods		
4	Title	**Applications of Explainable AI**:	1	6
	4.1	Visual Scene Understanding for Autonomous Driving Using Semantic Segmentation, Understanding Patch-Based Learning of Video Data by Explaining Predictions, Interpretable Deep Learning in Drug Discovery, Neural Hydrology – Interpreting LSTMs in Hydrology, Feature Fallacy, Current Advances in Neural Decoding: Complications with Interpreting Linear Decoding Weights in fMRI		
5*	Self Study			4
	5.1	Software and Application Patterns for Explanation Methods, Tensor Flow / Keras / SystemML / PyTorch **Practice Exercise**: explore Tensor Flow / keras / SystemML / PyTorch		
Total				**28**

*** Total 28 hrs does not include this module hrs.**

Laboratory Component (Minimum 10 Laboratory experiments are expected)

Sr. No	Title of the Experiment
1	Develop understanding of reverse engineering in NN
2	Text-to-Image Synthesis using MS-COCO dataset
3	Unsupervised Discrete Representation Learning
4	Perform sequential data modeling and forecasting using LSTM
5	Study of Software and application patterns for explanation methods

Text Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Explainable AI: Interpreting, Explaining and Visualizing Deep Learning,	LNAI, volume 11700	Samek, W., Montavon, G., Vedaldi, A., Hansen, L.K., Müller, K.-R. (Eds.)	Springer	2019

Course (Category) Code	Course Name	Teaching Scheme (Hrs/week)					Credits Assigned			
		L	T	P	O	E	L	T	P	Total
(PE)	Mobile Device Forensics & Cloud Forensics	2	0	2	3	7	2	0	1	3
		Examination Scheme								
		Component	ISE		MSE		ESE			Total
CS533		Theory	50		50		100			200
		Laboratory	50		--		50			100

Pre-requisite Course Codes, if any.	Cyber Security and Digital Forensics, Digital Forensics and Incident Response (DFIR)
Course Objective: To prepare students for cyber security and digital forensics job profiles in Cyber security Industry, Academia and Research.	
Course Outcomes (CO): ***At the End of the course students will be able to***	
CS533.1	Differentiate between computer and mobile device forensics.
CS533.2	Analyze requirements of mobile device forensics and setup mobile device forensics laboratory various tool.
CS533.3	Demonstrate techniques and tools used for mobile device forensics investigations with documents creation and maintain chain of custody.
CS533.4	Select appropriate tools for cloud storage forensics.
CS533.5	Prepare and present report on mobile device and cloud forensics as per security compliance

CO-PO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PO1	PO2	PO3
CS533.1			
CS533.2			
CS533.3			
CS533.4			
CS533.5			

CO-PEO/PSO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PEO1	PEO2
CS533.1		
CS533.2		
CS533.3		
CS533.4		
CS533.5		

BLOOM'S Levels Targeted (Pl. Tick appropriate)

Remember	Understand	Apply	Analyze	Evaluate	Create

Theory Component

Module No.	Unit No.	Topics	Ref.	Hrs.
1		**Introduction to Mobile Forensics :** Mobile Phone Basics, Inside Mobile device: Cell Phone Crime, SIM Card, SIM Security, Mobile forensic & its challenges,, Mobile phone evidence extraction process, Practical mobile forensic approaches, Mobile operating systems overview, Mobile forensic tool leveling system, Data acquisition methods. **Practical Exercise:** Popular tools for manual extractions include: Project-A-Phone Fernico ZRT and EDEC Eclipse The tools used for logical extraction include: XRY Logical , Oxygen Forensic Suite and Lantern	6,8	**6**
2		**Electronics Evidences:** Electronic evidence stored on mobile phone, Rules of evidence Good forensic practices, Securing the evidence, Preserving the evidence, Documenting the evidence **Practical Exercise:** The common tools used for hex dump include: XACT , Cellebrite UFED Physical Analyzer And Pandora's Box**.** The popular tools and equipment used for chip-off include: iSeasamo Phone Opening Tool , Xytronic 988D Solder Rework Station, FEITA Digital inspection station, Chip Epoxy Glue Remover, Circuit Board Holder	3,4	**6**
3		**The Android device model**: The Linux kernel layer, Libraries, Dalvik virtual machine, The application framework layer, The applications layer. Android security, Android file hierarchy, Android file system, Viewing file systems on an Android device, Extended File System – EXT, Android Forensic Setup and Pre Data Extraction Techniques A forensic environment setup, Screen lock bypassing techniques Gaining root access, Android Data Extraction Techniques, Imaging an Android Phone, Data extraction techniques,, Android Data Recovery Techniques, Data recovery, Android App Analysis. **Practical Exercises:**	10,11	**6**

		Overview of Forensic Tools, Android app analysis,Reverse engineering Android apps, Cellebrite – UFED, MOBILedit,, Autopsy, Digital Evidence from Smart band, smart watch, IoT devices.		
4		**Introduction to Cloud Forensics:** Cloud Computing Basics-Cloud Architecture-SAAS,IAAS & PAAS Cloud types-Public, Private, Community and Hybrid.Usage of cloud forensics, Challenges and Issues with Cloud Forensics,The three dimensions of cloud forensics, Cloud forensic organizational structure, Chain of Dependencies,Multi-Jurisdiction and multi-tenancy,Digital forensics evidence acquisition in cloud storage service: examining and evaluating tools and techniques, **Standards and Policies:** GDPR clauses and Cloud contract (IaaS, PaaS,SaaS)	5,7, 12	**6**
5*		**Self Study:** **Case Study:** 1. Forensics-as-a-Service 2. Cloud storage forensics: ownCloud as a case study **Practical Exercises:** Tools for Cloud Forensic : FROST , UFED Cloud Analyzer, . diffy, MD-CLOUD, EnCase, FTK, Oxygen Forensic, SIFT	13,14	**04**
			Total (* Not Included)	**28**

Laboratory Component (Minimum 10 Laboratory experiments are expected)

Sr. No	Title of the Laboratory Exercises
1	Setting up Mobile/Handheld Device Forensic Laboratory
2	Mobile/ Smart Phone Forensic Part-I
3	Mobile/ Smart Phone Forensic Part-II
4	Mobile/ Smart Phone Forensic Part-III
5	Handheld/Portable Device Forensics
6	Mobile Malware Analysis
7	Automated Forensic Analysis of Mobile Applications on Android Devices
8	Application Development for Mobile Forensics and Investigation
9	Tools for Cloud Forensic : FROST , UFED Cloud Analyzer, diffy, MD-CLOUD
10	Cloud storage forensics: own Cloud as a case study

Text Books:

Sr. No	Title	Authors	Publisher	Year
1	Crime Scene Forensics A Scientific Method Approach	Robert C Shaler	CRC Press	2011
2	Computer Forensics and Cyber Crime: An Introduction	Marjie T. Britz	Pearson	2013
3	Computer Forensics Evidence Collection & Preservation	EC-Council	Cengage Learning	2010
4	Digital Evidence and Computer Crime: Forensic Science, Computers, and the Internet	Eoghan Casey BS MA	Academic Press	2011
5	Computer Forensics: Computer Crime Scene Investigation	John R. Vacca	Charles River Media	2002
6	Malware Forensics: Investigating and Analyzing Malicious Code	Cameron H. Malin ,Eoghan Casey, James M. Aquilina	Syngress	2008
7	Introduction to Network Forensics	The European Union Agency for Network and Information Security (ENISA)	ENISA	2019
8	Practical Mobile Forensics	Satish Bommisetty and Rohit Tamma	PacktPublishingLimited	2014
9	Mobile phone security and forensics: A practical approach	Iosif I. Androulidakis	Springer publications	2012
10	Android Forensics: Investigation, Analysis and Mobile Security for Google Android	Andrew Hogg	Synergy	2011
11	iPhone and iOS Forensics: Investigation, Analysis and Mobile Security for Apple iPhone, iPad and iOS Devices	Andrew Hoog and Katie Strzempka	Synergy	2011
12	Wireless Crime and Forensic Investigation	Gregory Kipper	Auerbach Publications;	2007
13	Practical Cloud Security: A Guide for Secure Design and Deployment	Chris Dotson	Shroff O'Reilly	2019

Course (Category) Code	Course Name	Teaching Scheme (Hrs/week)					Credits Assigned			
		L	T	P	O	E	L	T	P	Total
(SBC)	High Performance Computing Lab	1	0	4	3	8	1	0	2	3
		Examination Scheme								
		Component	ISE		MSE		ESE			Total
CS508		Theory	--		--		--			--
		Laboratory	150		--		150			300

Pre-requisite Course Codes, if any.	1. Operating Systems 2. Design and Analysis of Algorithms 3. Computer Networks 4. Programming Language Concepts
Course Objective:-	
Course Outcomes (CO):*At the End of the course students will be able to*	
CS508.1	Analyze a programming task and identify what portions admit a parallel implementation.
CS508.2	Apply the different parallel computing approaches using MPI and OpenMP platform for achieving high performance.
CS508.3	Apply the different parallel computing approaches using OpenCL platform for achieving high performance.
CS508.4	Develop well-optimized threaded applications using memory management and data transfer methodology on CUDA platform for achieving high performance.

CO-PO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PO1	PO2	PO3
CS508.1			
CS508.2			
CS508.3			
CS508.4			

CO-PEO/PSO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PEO1	PEO2
CS508.1		
CS508.2		
CS508.3		
CS508.4		

BLOOM'S Levels Targeted (Pl. Tick appropriate)

Remember	Understand	Apply	Analyze	Evaluate	Create

Theory Component

Module No.	Unit No.	Topics	Ref.	Hrs.
1	Title	**HPC Architectures and Algorithm Design**	1,2	3
	1.1	**Standard Parallel Programming Systems**: MPI & OpenMP, OpenCL, CUDA, Heterogeneous Computer Memories and Data Transfer, Host Code, Applications of Heterogeneous Computing, Benchmarking CGM		
	1.2	**Parallel Algorithm Design:** Task/Channel Model, Foster's Design Methodology, Boundary Value Problem, Finding the Maximum, The n-Body Problem		
2	Title	**MPI and OpenMP Programming**	2	2
	2.1	**Message-Passing Programming:** Message-Passing Model, Message-Passing Interface, Circuit Satisfiability Problem, Collective Communication		
	2.2	**Shared-Memory Programming:** Shared-Memory Model, Parallel for Loops, Declaring Private Variables, Critical Sections, Reductions		
3	Title	**OpenCL Fundamentals and Programming**	1	3
	3.1	**OpenCL Overview:** Using OpenCL, Platforms and Devices, OpenCL Platforms C++, OpenCL Context to Manage Devices, OpenCL Context to Manage Devices using C++, Error Handling		
	3.2	**Work-Item and Work-Groups:** Command Queues, Work-Items and Work-Groups, OpenCL Memory, Programming and Calling Kernel.		
	3.3	**OpenCL Example:** Structure of the OpenCL Host Program, Structure of OpenCL host Programs in C++, The SAXPY Example, Step by Step Conversion of an Ordinary C Program to OpenCL		
4	Title	**CUDA GPU Programming - I**	3	3
	4.1	**CUDA Basics -** Timing the Kernel, Timing with CPU Timer and nvprof, Organizing Parallel Threads, Managing Devices.		
	4.2	**CUDA Execution Model-** Nature of Warp Execution, Exposing Parallelism, Checking Active Warps, Memory Operations, Avoiding Branch Divergence, Unrolling Loops, Dynamic Parallelism		
5	Title	**CUDA GPU Programming - II**	3	3

	5.1	**CUDA Global, Shared and Constant Memory Model:** Memory Management, Memory Access Patterns, Matrix Addition with Unified Memory, Data Layout of Shared Memory, Reducing and Coalescing Global Memory Accesses, Constant Memory.		
	5.2	**CUDA Streams and Concurrency:** Streams and Events, Concurrent Kernel Execution, Overlapping Kernel Execution and Data Transfer, Overlapping GPU and CPU Execution.		
6	**Self Study**	1) Parallel Architectures: Interconnection Networks, Processor Arrays, Multiprocessor, Multicomputer, Flynn's Taxonomy. 2) Investigate the various GPU-accelerated CUDA libraries like CUSPARSE, cuBLAS, cuFFT and cuRAND library.	3	4*
			Total (* Not Included)	14

Laboratory Component (Minimum 10 Laboratory experiments are expected)

Sr. No.	Title of the Experiment
	MPI Programming
1	Implement Sieve of Eratosthenes with following data decomposition options: i) interleaved data decomposition and ii) block data decomposition using Parallel MPI programming and then analyze the algorithm.
2	Implement Floyd's version of All-Pair Shortest-Paths Problem through four steps of parallel algorithm design namely partitioning, communication, agglomeration and mapping decomposition using Parallel MPI programming and then analyze the algorithm.
3	Implement Matrix-Vector Multiplication using various Data Decomposition Options: i) Rowwise Block-Striped, ii) Columnwise Block-Striped Decomposition and iii) Checkboard Block Decomposition using Parallel MPI programming and then analyze the algorithm.
4	Implement Matrix Multiplication algorithm through i) Sequential Algorithm, ii) Parallel Algorithm - a) Rowwise Block-Striped Decomposition and b) Cannon's Algorithm using Parallel MPI programming and then analyze the algorithm.
5	Implement Linear Systems solution using i) Back substitution, ii) Gaussian Elimination, iii) Iterative Method and iv) Conjugate Gradient Method using Parallel MPI programming and then analyze the algorithm.
	MPI and OpenMP Programming
6	Implement Linear Systems solution using i) Conjugate Gradient Method and ii) Jacobi Method using Parallel MPI and OpenMP programming and then analyze the algorithm.
	OpenCL Programming
7	A histogram is a statistic that shows the frequency of a certain occurrence within a data set. The histogram of an image provides a frequency distribution of pixel values in the image. Write an OpenCL parallel implementation of the histogram algorithm by breaking the image into tiles, compute the histogram for each tile, and then combine the partial histograms computed for each tile into the final histogram of the image. If the image is a

	color image, the pixel value can be the luminosity value of each pixel or the individual R, G, and B color channels.
8	Write an OpenCL parallel implementation for Dijkstra's Single-Source Shortest-Path Graph Algorithm.
	CUDA Programming
9	Write a CUDA C program to find Matrix Transpose with i) shared memory, ii) padded shared memory and iii) unrolling for a large number of rows and columns of matrix.
10	Write a CUDA C program to integrate function using Trapezoidal Rule on GPU through Synchronization and concurrent execution.

Text Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	Using OpenCL_ Programming Massively Parallel Computers	First	Janusz Kowalik and Tadeusz Puzniakowski	IOS Press	2012
2	Parallel programming in C with MPI and Open MP	First	Michael J Quinn	McGraw-Hill	2003
3	Professional CUDA C Programming	First	John Cheng, Max Grossman, Ty McKercher	Wrox	2014

Reference Books

Sr. No	Title	Edition	Authors	Publisher	Year
1	OpenCL Programming Guide	First	Aaftab Munshi, Benedict R. Gaster, Timothy G. Mattson, James Fung, Dan Ginsburg	Addison-Wesley	2012
2	OpenCL Parallel Programming Development Cookbook-	First	Raymond Tay	Packt Publishing	2013
3	CUDA by Example : An Introduction to General-Purpose GPU Programming	First	Jason Sanders, Edward Kandrot	Addison-Wesley	2010

Course (Category) Code	Course Name	Teaching Scheme (Hrs/week)					Credits Assigned			
		L	T	P	O	E	L	T	P	Total
(SBC)	Enterprise Technology Laboratory	0	0	2	3	5	0	0	1	1
		Marking Scheme								
		Component		ISE		MSE		ESE		Total
CS509		Theory		--		--		--		--
		Laboratory		50		--		50		100

Pre-requisite Course Codes	
Course Outcomes: ***At the End of the course students will be able to***	
CS509.1	Build core skill sets required for designing and deploying dynamically scalable, highly available, fault-tolerant, and reliable server less applications on Public Cloud Infrastructure.
CS509.2	Understand DevOps techniques and implement tools for enterprise automation to deal with complexity.
CS509.3	Build a CI/CD pipeline using various DevOps tools and deploy an application on cloud platform.

CO-PO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PO1	PO2	PO3
CS509.1			
CS509.2			
CS509.3			

CO-PEO/PSO Correlation Matrix (3-Strong, 2-Moderate, 1-Weak Correlation)

	PEO1	PEO2
CS509.1		
CS509.2		
CS509.3		

BLOOM'S Levels Targeted (Pl. Tick appropriate)

Remember	Understand	Apply	Analyze	Evaluate	Create

<table>
<tr><th>Exp. No.</th><th>Experiment Details</th><th>Ref.</th></tr>
<tr><td rowspan="2">1</td><td>Title: AWS Fundamentals: Going Cloud Native</td><td></td></tr>
<tr><td>1.1. Deploy a multi-tier website on AWS
Problem Statement: Deploying a Custom PHP Website to AWS with functionalities for SQL, NoSQL and file storage
Topics – RDS, SNS, DynamoDB, S3, VPC, EC2, NAT Gateways, Load Balancer and Auto Scaling

1.2. Deploying a website for High Availability and High Resilience
Problem Statement: Design an architecture which can automatically scale up and down based on traffic and is de coupled for components like Database, webapp etc.
Topics: Auto Scaling, Target Groups, Load Balancing, RDS, PaaS, ElasticBeanstalk</td><td>1 ,6</td></tr>
<tr><td>2</td><td>1.3. Demonstrate skills in building and deploying serverless solutions on AWS Cloud platform using range of AWS services
Problem Statement: Using application services in AWS to deploy code, configuration management using OpsWorks
Topics: Elastic Beanstalk, AWS Lambda, SNS, Amazon API Gateway, Amazon DynamoDB, and Amazon Lex.</td><td>3</td></tr>
<tr><td rowspan="3">3</td><td>Title: DevOps: Using enterprise automation to deal with complexity</td><td></td></tr>
<tr><td>2.1.Microservices From Development to Production Using Docker, Docker Compose & Docker Swarm:
Problem statement: Design a Microservices Architecture for a full stack application; containerize various microservices using Docker Compose and Docker Swarm.</td><td>2</td></tr>
<tr><td>2.2. Continuous Integration using Jenkins by building and automating test cases using Maven.
Problem statement: Build the pipeline of jobs using Jenkins, create a pipeline script to deploy an application over the tomcat server</td><td>2,4</td></tr>
<tr><td></td><td>2.3. Demonstrate Infrastructure as a code/Configuration as a code using configuration management tools like Ansible/Puppet/Chef/Salt stack.
Problem statement: Install and configure one of the configuration management tools and implement servers using it.</td><td>5</td></tr>
<tr><td>4</td><td>Title: Mini Project : based on the experiments done on topics 1 and 2
Evaluation of Mini-Project will be based on the following rubrics :
<table><tr><td>Performance Indicator</td></tr><tr><td>Identification of problem</td></tr><tr><td>Justification of technology stack</td></tr><tr><td>Implementation</td></tr><tr><td>Presentation</td></tr><tr><td>Report Writing</td></tr></table></td><td></td></tr>
</table>

Textbooks

Sr. No	Title	Edition	Authors	Publisher	Year
1	AWS Certified Solutions Architect Study Guide, Sybex	Second	Ben Piper, David Clinton	Wiley	2019
2	Docker Up & Running: Shipping Reliable Containers in Production	First	Sean P. Kane, Karl Matthias	O'Reilly Publication	2015

Reference Books

Sr. No	Title	Edition	Authors	Publisher	Year
3	Beginning Serverless Computing: Developing with Amazon Web Services, Microsoft Azure, and Google	First	Maddie Stigler	Apress	2017
4	Continuous Delivery with Docker and Jenkins	First	Rafal Leszko	Packt Publication	2017
5	Ansible: Up and Running- Automating Configuration Management and Deployment the Easy	Second	Lorin Hochstein	O'Reilly Publication	2017
6	Online Resource https://docs.aws.amazon.com/servicecatalog/				