Integral University, Lucknow

Effective from Session: 2016-17							
Course Code	CS-301	**Title of the Course**	Design and Analysis of Algorithm	**L**	**T**	**P**	**C**
Year	III	**Semester**	V	3	1	0	4
Pre-Requisite	None	**Co-requisite**	None				
Course Objectives	To analyze the problem and design an efficient algorithm to solve it by using & modifying classical design techniques or creating a new solution technique.						

Course Outcomes	
CO1	Would be able to analyze the problem and design an efficient algorithm to solve it by using & modifying classical design techniques or creating a new solution technique.
CO2	For an algorithm given all the required parameters, would be able to analyze the algorithm and evaluate its utility in the given situation, able to apply the approach where problem can be solved by smaller input then apply for larger perspective
CO3	Given more than one solution for the problem, would be able to evaluate and compare those using standard mathematical techniques and select the best solution.
CO4	For a design problem given, would be able to compare and evaluate different Data Structures available and modify or create new them for the same.
CO5	For given different problems, would be able to categorize the different kind of complexities and develop non deterministic solution to problems having large complexities.

Unit No.	**Title of the Unit**	**Content of Unit**	**Contact Hrs.**	**Mapped CO**
1	**Introduction:**	Introduction: Algorithms, Analysis of Algorithms, Growth of Functions: Asymptotic Notations, Standard Notations and Common Functions; Recurrence Methods: Substitution Method, Iteration Method, Recursion Tree Method, Master's Method.	8	1
2	**Designing of Algorithms and Advanced Data Structure**	Divide & Conquer: Heap Sort, Quick Sort, Sorting in Linear Time, Medians and Order Statistics. Red-Black Trees, Augmenting Data Structure, Binomial Heaps, Fibonacci Heaps.	8	2
3	**Advanced Design and Analysis Techniques**	Greedy Algorithms: Knapsack Problem, Travelling Salesperson Problem, Minimum Cost Spanning Trees: Kruskal's Algorithm, Prim's Algorithm. Dynamic Programming: Longest Common Subsequence, Matrix Chain Multiplication, 0/1 Knapsack Problem, Single Source Shortest Path: Dijkstra's Algorithm, Bellman Ford Algorithm.	8	3
4	**Amortized Analysis, Back Tracking: and Branch & Bound**	Accounting Method, Aggregate Method, Potential Method, Introduction, Subset Sum Problem, n-Queens problem and Introduction, 0/1 Knapsack, 15 Puzzle problem.	8	4
5	**String Matching and Complexity Theory**	Algorithm, The Rabin-Karp Algorithm, The Knuth-MorrisPratt Algorithm. Class P and NP, NP-hard Problems, NP-Complete Problems, Polynomial Reduction, Approximation Algorithm	8	5

Reference Books:

1. Coremen, Rivest, Lisserson, "Algorithms", PHI.
2. Horwitz & Sahani, Fundamental of Computer Algorithm, Galgotia.
3. Michael T. Goodrich and Roberto Tamassia, Algorithm Design: Foundation, Analysis and Internet Examples, John Wiley Publications.

e-Learning Source:

1. https://onlinecourses.nptel.ac.in/noc19_cs47/preview
2. https://nptel.ac.in/courses/106106131

	Course Articulation Matrix: (Mapping of COs with POs and PSOs)														
PO-PSO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	1	3	3	1	3		3	1	1			1	1	3	
CO2	2	2	3	3	1	1	2	2	1			2	2	2	1
CO3	1	1	1	2	3	1		2	2			1		3	1
CO4	2	2	1	2	2	1	2	1	3		1		2	1	1
CO5	1	2	1	3	1		1		2	3	1	1	1	2	3

1- Low Correlation; 2- Moderate Correlation; 3- Substantial Correlation

Integral University, Lucknow

Effective from Session: 2016-17							
Course Code	CS-303	**Title of the Course**	Principles of Operating System	**L**	**T**	**P**	**C**
Year	III	**Semester**	V	3	1	0	4
Pre-Requisite	None	**Co-requisite**	None				
Course Objectives	To introduce students with basic concepts of Operating System, its functions and services. To critique how memory management is implemented by the operating system, including concepts of paging, segmentation, paged segmentation etc. To introduce the concepts of Processes in Operating System and various algorithms to schedule these processes. To provide the knowledge of basic concepts towards process synchronization, deadlock and related issues. To gain insight on file management, disk management etc and to become familiar with the protection and security mechanisms taken by operating system.						

Course Outcomes	
CO1	The basic concepts of Operating System, its functions and services.
CO2	Design and effective memory management scheme for the operating system where there is less wastage and the response time is quick.
CO3	The basic concepts of Processes in Operating System and the application of various CPU scheduling algorithms.
CO4	Analyse the basic concepts of process synchronization, deadlock and related issues.
CO5	The basic components of file management, disk management etc and will become familiar with the protection and security mechanisms taken by operating system.

Unit No.	Title of the Unit	Content of Unit	Contact Hrs.	Mapped CO
1	**Desktop OS and Mobile OS**	Importance of Operating Systems; Basic Concepts and Terminology; Evolution of Operating Systems: Batch, Interactive, Time Sharing & Real Time Systems. Operating System Structure: Simple Structure, Layered Approach; System Calls; Kernels: overview, objectives of kernel, types of kernels. Architecture, Android OS, iOS, Virtual OS, Cloud OS and their design.	8	1
2	**Process, Threads, CPU Scheduling and Real Time Scheduling**	Introduction, Process Model, Process State, Process Control Block. Overview, benefits of threads, types of threads. Basic Concepts, Scheduling Criteria, And Types of Scheduling, Scheduling Algorithms: FCFS, SJF, Round Robin, Priority Scheduling, Multilevel Queue Scheduling, Multilevel, Feedback Scheduling. Introduction, Uniprocessor scheduling, Multiprocessor Scheduling.	8	2
3	**Process Synchronization and Deadlock**	Principles of Concurrency, Race Condition, Critical Section, Critical Section Problem, Synchronization Mechanism, Semaphores and Classical Problems of Synchronization: Bounded Buffer Problem, Readers Writers Problem. Principles, System Model, Deadlock Characterization, Methods of Deadlock Handling: Prevention, Avoidance, Detection & Recovery from Deadlock	8	3
4	**Memory Management and Virtual Memory Management**	Introduction, logical vs. physical address space, swapping, contiguous memory allocation, paging, segmentation, segmentation with paging. Introduction, demand paging, performance, page replacement, page replacement algorithms (FCFS, LRU, Optimal), allocation of frames, thrashing. Other Memory Management Schemes: Swapping, Overlays.	8	4
5	**Device Management, Disk Scheduling and Protection & Security**	Introduction, types of devices, FCFS, SSTF, SCAN, CSCAN, LOOK, C-LOOK Scheduling File Systems: file concept, Access Mechanism, directory structure, file system structure, allocation methods (Contiguous, linked, indexed), free-space management (bit vector, linked list, grouping), Directory implementation (linear list, hash table), efficiency & performance.	8	5

Reference Books:

1. Galvin, Silberchatz "Operating Systems Principles", Addision Wesley.
2. Milenekovie, "Operating System Concept", McGraw Hill.
3. Dietal, "An Introduction to Operating System", Addion Wesley.
4. Tannenbaum, "Operating System Design And Implementation", PHI.
5. Galvin, Silberchatz "Operating Systems Principles", Addision Wesley.

e-Learning Source:

1. https://nptel.ac.in/courses/106105214

	Course Articulation Matrix: (Mapping of COs with POs and PSOs)														
PO-PSO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	1	1	3	2	3				1		3			
CO2	3	3	3	2	1	1		1				2			
CO3	3	2	1	1	2	2	3		2			3			
CO4	3	2	2	2	3	3				1		2			
CO5	3	1	1	1	1	2	1					2			

1-Low Correlation; 2- Moderate Correlation; 3- Substantial Correlation

Integral University, Lucknow

Effective from Session: 2016-17							
Course Code	CS-304	**Title of the Course**	Theory of Automata & Formal Languages	**L**	**T**	**P**	**C**
Year	III	**Semester**	V	3	1	0	4
Pre-Requisite	None	**Co-requisite**	None				
Course Objectives	The primary objective of this course is to introduce students to the foundations of computability theory. Other objectives include the application of mathematical techniques and logical reasoning to important problems, and to develop a strong background in reasoning about finite state automata and formal languages.						

Course Outcomes	
CO1	To demonstrate computational mathematical models for problem solving and describe how they relate to formal languages.
CO2	To analyse the relationship among language classes and grammars with the help of Chomsky Hierarchy.
CO3	To apply rigorous formal mathematical model for proving different properties of grammars, languages and automata.
CO4	To apply mathematical foundations, algorithmic principles and computer science theory to the modelling and design of computer based systems in a way that demonstrates.
CO5	Have an overview of how the theoretical study in this course is applicable to and engineering application like designing the compilers.

Unit No.	**Title of the Unit**	**Content of Unit**	**Contact Hrs.**	**Mapped CO**
1	**Finite State Machines**	Finite state machine, definitions, Finite automaton model, acceptance of strings and languages Deterministic Finite Automata (DFA), Nondeterministic Finite Automata (NFA), NFA with epsilon transition, Language of NFA, Equivalence of NFA and DFA, Minimization of Finite Automata, FA with output: Moore and Mealy machine, Equivalence of Moore and Mealy Machine, Applications and Limitation of FA..	8	1
2	**Regular Languages**	Chomsky Hierarchy, Regular Grammars, Unrestricted Grammars, Context Sensitive Language, Regular expression (RE); Definition, Operators of regular expression and their precedence, Algebraic laws for Regular expressions, Kleen's Theorem, Regular expression to FA, DFA to Regular expression, Arden Theorem, Non-Regular Languages, Pumping Lemma for regular Languages. Application of Pumping Lemma, Closure properties of Regular Languages, Decision properties of Regular Languages	8	2
3	**Context Free Grammar**	Context free grammar (CFG) and Context Free Languages (CFL): Definition, Examples, Derivation, Derivation trees, Ambiguity in Grammar, Ambiguous to Unambiguous CFG, Useless symbols, Simplification of CFGs, Normal forms for CFGs: CNF and GNF, Closure properties of CFLs, Decision Properties of CFLs: Emptiness, Finiteness and Membership, Pumping lemma for CFLs.	8	3
4	**Push Down Automata**	Push Down Automata (PDA): Description and definition, Instantaneous Description, Language of PDA, Acceptance by Final state, Acceptance by empty stack, Deterministic PDA, Equivalence of PDA and CFG, CFG to PDA and PDA to CFG, Two stacks PDA, Non-Deterministic Push Down Automata.	8	4
5	**Turing Machines**	Turing machines (TM): Basic model, definition and representation, Instantaneous Description, Language acceptance by TM, Variants of Turing Machine, TM as Computer of Integer functions, Universal TM, Church's Thesis, Recursive and recursively enumerable languages, Halting problem, Introduction to Undecidability, Undecidable problems about TMs. Post correspondence problem (PCP), Modified PCP, Introduction to recursive function theory	8	5

Reference Books:

1. Hopcroft and Ullman, "Introduction to Automata Theory Languages and Computation", Addison Wesley.
2. Mishra & Chandrasekhar, "Theory of Computer Sciences", PHI.
3. Peter Linz, "An Introduction to Formal Languages and Automata", Jones & Bartlett Learning. Recommended Prerequisite – CS206Co-requisite - None

e-Learning Source:

1. https://nptel.ac.in/courses/106105196

	Course Articulation Matrix: (Mapping of COs with POs and PSOs)														
PO-PSO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	2	2	2		1		1		1		2	1		3
CO2	3	2	3	2	2	1						2		3	
CO3	3	3	3	2	1	2			2			2	2		3
CO4	3	2	2	2	3	3	1	2		1	1	2	2		
CO5	3	2	1	1		2				2		2		1	2

1-Low Correlation; 2- Moderate Correlation; 3- Substantial Correlation

Integral University, Lucknow

Effective from Session: 2016-17							
Course Code	CS-340	**Title of the Course**	Software Engineering	**L**	**T**	**P**	**C**
Year	III	**Semester**	V	3	1	0	4
Pre-Requisite	None	**Co-requisite**	None				
Course Objectives	1. Explain the basic understanding of software, its characteristics, and importance of following engineering principles to develop software. 2. Assess the applicability, strengths, and weaknesses of the different development life cycle models to provide real world software solutions. 3. To understand various processes of each phase of SDLC and make the students capable to prepare quality documentation for software development. 4. To develop effort estimation and risk management skills for developing software. **5.** Study of CASE tools, Quality Assurance activities etc. for focusing on quality issues of software.						

Course Outcomes	
CO1	Identify the best suitable SDLC model for a given set of user requirements.
CO2	Estimate the total effort, to assess and manage the potential risks involved while developing the software.
CO3	Create a good quality SRS and design a highly cohesive and low coupled software.
CO4	Follow the standard coding guidelines and practices and prepare best possible test cases to uncover errors.
CO5	Work on modern CASE tools and follow the international quality standards to produce good quality software.

Unit No.	Title of the Unit	Content of Unit	Contact Hrs.	Mapped CO
1	**Introduction to Software Engineering**	Types of Software, Software Characteristics, Quality of a Good Software, Software Myths, Software Components, Software Crisis, Software Engineering: Definition, Challenges, Software Engineering Processes, Similarity and Differences from Conventional Engineering Processes. Software Development Life Cycle Models: Build and Fix Models, Waterfall Model, Prototyping Model, RAD Model Iterative Enhancement Model, Evolutionary Development Model and Spiral Model, WINWIN Spiral Model, Fourth Generation Techniques.	8	1
2	**Planning a Software**	Process Planning, Effort Estimation: Uncertainities in Effort Estimation, Building Effort Estimation Models, A Bottom-Up Estimation Approach, COCOMO Model, Project Scheduling & Staffing: Overall Scheduling, Detailed Scheduling, Team Structure, Software Configuration Management(SCM): - Baselines, Version Control, Change Control & Configuration Audit, Risk Management: Reactive and Proactive Risk Strategies, Software Risks, Risk Analysis, Identification, Projection, Assessment, Monitoring and Managing the Risk, RMMM Plan.	8	2
3	**Software Requirements Analysis and Specification**	Software Requirements: Need for SRS, Requirement Process, Problem Analysis: Informal & formal Approaches, Data Flow Modeling, Object Oriented Modeling, Prototyping, Requirements Specifications: Characteristics of an SRS, Components of SRS, Specification Language, Structure of Requirement Document: IEEE Standards for SRS, Validation, Metrics. **Designing and Coding:** Designing: Function Oriented Design: Design Principles: Problem Partitioning and Hierarchy, Abstraction, Modularity, Top Down and Bottom-Up Strategies, Module Level Concepts: Coupling, Cohesion; Structure Design Methodology, Verification, Introduction to Object Oriented Design & User Interface Design, Software Measurement Metrics: Various Size Oriented Measures- Halestead's Software Science, Function Point (FP) Based Measures, Cyclomatic Complexity Measures: Control Flow Graphs.	8	3
4	**Coding & Testing**	**Coding:** Programming Principles and Guidelines: Common Coding Errors, Structured Programming, Information Hiding, Programming Practices, Coding Standards, Coding Process, Refactoring, Verification: Code Inspection, Static Analysis, Proving Correctness, Combining Different Techniques, Metrics. **Testing:** **Testing Fundamentals:** Error Fault and Failure, Test Oracles, Test Cases and Test Criteria, Test Case Execution and Analysis, Unit Testing, Integration Testing: : Top Down and Bottom up, Acceptance Testing: Alpha and Beta Testing., Regression Testing, functional and non-functional testing. Testing Techniques: White Box: Logic Coverage, Path Coverage, Loop Coverage, Data Flow Testing. Black Box Testing: Boundary Value Analysis, Equivalence Class Testing, state Table Based Testing, Decision Table Based Testing.	8	4
5	**Computer Aided Software Engineering (CASE)**	CASE Tools, Scope, Benefits of CASE Tool, support in Software Life Cycle, Architecture of CASE Environment, Types of CASE Tools, Software Reliability and Quality Management: -Software Quality Management: Quality Concepts, Software Quality Assurance, Software Reviews, Formal Technical Reviews, and Statistical Quality Assurance. Software Reliability, ISO 9000 Quality Standards, CMM Levels.	8	5

Reference Books:

1. Software Engineering: A Practitioner's Approach by Roger S. Pressman, McGraw-Hill International edition.
2. An Integrated Approach to Software Engineering, by Pankaj Jalote, Narosa Publishing House.
3. Software Engineering by K.K. Agarwal.
4. Software Engineering by Ian Sommerville, Addison-Wesley.

5. Fundamentals of Software Engineering by Rajib Mall, PHI.

e-Learning Source:

1. https://onlinecourses.nptel.ac.in/noc20_cs68/preview

	Course Articulation Matrix: (Mapping of COs with POs and PSOs)															
PO-PSO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CO1	2	3	2	2		1	1	2	2	3	2	1	3			
CO2	3	3	2	2	1		1		2	1	3	1		2	1	
CO3	3	3	3	2					3	3		2		3		
CO4	3	2	2	2	1	3		2	3	3		2	2	1		
CO5	3	1	3	2	3	2	1	2	2	2	2	2			1	1

1-Low Correlation; 2- Moderate Correlation; 3- Substantial Correlation

Integral University, Lucknow

Effective from Session: 2016-17							
Course Code	CS-342	**Title of the Course**	DATA COMPRESSION	**L**	**T**	**P**	**C**
Year	III	**Semester**	V	3	1	0	4
Pre-Requisite	None	**Co-requisite**	None				
Course Objectives	• Basic knowledge of Data compression • Types of data compression • Various techniques of Data compression • Application of data compression						

Course Outcomes	
CO1	Understand the importance of compressions, and different compression models
CO2	Solve the various problems based on lossless compression approach such as Huffman, adaptive Huffman models
CO3	Solve problems using arithmetic and dictionary-based compression techniques.
CO4	Apply partial prediction matching, and learn to transformation of source based on Transform algorithms
CO5	Represent the various dynamic model in the form of structured vector representation

Unit No.	**Title of the Unit**	**Content of Unit**	**Contact Hrs.**	**Mapped CO**
1	**Introduction:**	Compression Techniques: Loss Less Compression, Lossy Compression, Measures of Performance, Modeling and Coding. Mathematical Preliminaries for Lossless Compression: A Brief Introduction to Information Theory: - Models: Physical Models, Probability Models, Markov Models, Composite Source Model, Coding: -Uniquely Decodable Codes, Prefix Codes	8	1
2	**Huffman Coding**	The Huffman Coding Algorithm: Minimum Variance Huffman Codes, Adaptive Huffman Coding: Update procedure, Encoding procedure, decoding procedure. Golomb Codes, Rice Codes, Tunstall codes. Application of Huffman Coding. Text compression, Audio Compression.	8	2
3	**Arithmetic Coding**	Coding a Sequence, Generating a Binary Code, Comparison of Binary and Huffman Coding, Applications: Bi-Level Image Compression-JBIG and JBIG2 Standards. Dictionary Techniques: Introduction, Static Dictionary: - Diagram Coding, Adaptive Dictionary: The LZ77 Approach, The LZ78 Approach Applications. Image Compression: The Graphics Interchange Format (GIF), Compression over Modem.	8	3
4	**Prediction with Partial Match**	The Basic Algorithm, The ESCAPE SYMBOL, Length of Context, The Exclusion Principle, The Burrows-Wheeler Transform, Move-to- Front Coding, CALlC, JPEG-LS, Multi-resolution Approaches, Facsimile Encoding, Dynamic Markov Compression.	8	4
5	**Quantization**	Introduction of Scalar and Vector Quantization, Advantages of Vector Quantization Over Scalar Quantization, The Linde-Buzo-Gray Algorithm, Tree Structured Vector Quantizes, Structured Vector Quantizes.	8	5

Reference Books:

1. Introduction to Data Compression, Second Edition, Khalid Sayood, The Morgan Kaufmann Series
2. Data Compression: The Complete Reference 4th Edition by David Salomon, Springer
3. Text Compression1st Edition by Timothy C. Bell Prentice Hall
4. Elements of Data Compression, Drozdek, Cengage Learning

	Course Articulation Matrix: (Mapping of COs with POs and PSOs)														
PO-PSO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	2	2								3	2	2
CO2	2	2	2	2	2								2	2	1
CO3	2	2	1	1	2								3	2	1
CO4	3	2	1	2	1	2	1						3	2	1
CO5	2	2	3	2	2	1	1	1					3	2	2

1-Low Correlation; 2- Moderate Correlation; 3- Substantial Correlation

Integral University, Lucknow

Effective from Session: 2021-22							
Course Code	CS-391	**Title of the Course**	Hadoop	**L**	**T**	**P**	**C**
Year	III	**Semester**	V	3	1	0	4
Pre-Requisite	None	**Co-requisite**	None				
Course Objectives	1. To provide an overview of an exciting field of big data analytics and Hadoop. 2. Develop an understanding of the complete open-source Hadoop ecosystem and its near-term future direction. 3.To introduce the tools required to manage and analyze big data like Hadoop, MapReduce 4. To teach the fundamental techniques and principles in achieving big data analytics with scalability and streaming capability. 5.To enable students to have skills that will help them to solve complex real-world problems in decision support						

Course Outcomes	
CO1	To provide an overview of an exciting field of big data analytics and Hadoop
CO2	Develop an understanding of the complete open-source Hadoop ecosystem and its near-term future direction.
CO3	To introduce the tools required to manage and analyze big data like Hadoop, MapReduce
CO4	To teach the fundamental techniques and principles in achieving big data analytics with scalability and streaming capability.
CO5	To enable students to have skills that will help them to solve complex real-world problems in decision support.

Unit No.	Title of the Unit	Content of Unit	Contact Hrs.	Mapped CO
1	**BIG DATA**	Develop an understanding of the complete open-source Hadoop ecosystem and its near-term future directions, compare and evaluate the major Hadoop distributions and their ecosystem components both their strengths and their limitations, and hands-on experience with key components of various big data ecosystem components and roles in building a complete big data, Future of Big Data. Knowledge of data, How to use Big insight	9	1
2	**HADOOP**	Why Hadoop? What is Hadoop? Hadoop vs RDBMS, Hadoop vs Big Data, Types of Data, Brief history of Hadoop, Problems with traditional large-scale systems, Requirements for a new approach, Anatomy of a Hadoop cluster.	8	2
3	**HDFS**	Concepts & Architecture, Data Flow (File Read, File Write), Fault Tolerance, Shell Commands, Java Base API, Data Flow Archives, Coherency, Data Integrity, Role of Secondary Name Node, Zookeeper	8	3
4	**MAPREDUCE**	Theory, Data Flow (Map – Shuffle - Reduce), Map Red vs MapReduce APIs, Programming Mapper, Reducer, Combiner, Partitioner, Implementation of Mahout, R, Sqoop, Yarn, what is flume Flume, the architecture of Flume, Flume Modes, the overall architecture of Ambari and Ambari' relation to other services and components of a Hadoop cluster, the functions of the main components of Ambari, initiating start and stop services from Ambari Web Console	8	4
5	**HIVE AND PIG**	List the characteristics of representative data file formats including flat/text files CSV XML JSON and YAML, Architecture, Installation, Configuration, Hive vs RDBMS, Tables, DDL & DML, Partitioning & Bucketing, Hive Web Interface, Why Pig, Use case of Pig, Pig Components, Data Model, Pig Latin.	8	5

Reference Books:

1. Gelman, Andrew, and Jennifer Hill. Data Analysis Using Regression and Multilevel/Hierarchical Models. 1st ed. Cambridge, UK: Cambridge University Press, 2006. ISBN:9780521867061.
2. Gelman, Andrew, John B. Carlin, Hal S. Stern, and Donald B. Rubin. Bayesian Data Analysis. 2nd ed. New York, NY: Chapman & Hall, 2003. ISBN:9781584883883
3. Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data" by EMC Education Services
4. Analytics: Data Science, Data Analysis and Predictive Analytics for Business" by Daniel Covington.
5. Machine Learning for Big Data: Hands-On for Developers and Technical Professionals" by Jason Bell.

	Course Articulation Matrix: (Mapping of COs with POs and PSOs)														
PO-PSO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1		2				3	2					2			
CO2		2	3	2				1			2				
CO3	3	2	3	2	2				3		2				
CO4		2	3	3	3				2		3	3			
CO5	1	2	3	3	2		2		2		3	2			

1-Low Correlation; 2- Moderate Correlation; 3- Substantial Correlation

Integral University, Lucknow

Effective from Session: 2021-22							
Course Code	CS-395	**Title of the Course**	Predictive Analysis	**L**	**T**	**P**	**C**
Year	III	**Semester**	V	3	1	0	4
Pre-Requisite	none	**Co-requisite**	none				
Course Objectives	• How Statistical Modeling relates to Machine Learning and do a comparison of each. • Real-life examples of Machine learning and how it affects society in ways you may not have guessed! • In the labs: Use Python libraries for Machine Learning, such as scikit-learn. • To enable students to know real world implementation on Popular algorithms: Regression, Classification, and Clustering • To enable students about Recommender Systems: Content-Based and Collaborative Filtering						

Course Outcomes	
CO1	To enable students how Statistical Modeling relates to Machine Learning and do a comparison of each.
CO2	Real-life examples of Machine learning and how it affects society in ways you may not have guessed!
CO3	In the labs: Use Python libraries for Machine Learning, such as scikit-learn.
CO4	To enable students to know real world implementation on Popular algorithms: Regression, Classification, and Clustering
CO5	To enable students about Recommender Systems: Content-Based and Collaborative Filtering

Unit No.	**Title of the Unit**	**Content of Unit**	**Contact Hrs.**	**Mapped CO**
1	INTRODUCTION TO MACHINE LEARNING	• Applications of Machine Learning • Supervised vs Unsupervised Learning • Python libraries suitable for Machine Learning	9	1
2	REGRESSION	• Linear Regression • Non-linear Regression • Model evaluation methods	8	2
3	CLASSIFICATION	• K-Nearest Neighbor • Decision Trees • Logistic Regression • Support Vector Machines • Model Evaluation	8	3
4	UNSUPERVISED LEARNING	• K-Means Clustering • Hierarchical Clustering • Density-Based Clustering	8	4
5	RECOMMENDER SYSTEMS	• Content-based recommender systems Collaborative Filtering	8	5

Reference Books:

1. Machine Learning by Tom M. Mitchell
2. Python Machine Learning by Sebastian Raschka and Vahid Mirjalili
3. Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Technique to Build Intelligent Systems by Aurélien Géron
4. Understanding Machine Learning by Shai Shalev-Shwartz and Shai Ben-David La

PO-PSO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1		2	2	3		3						2			
CO2		2	3	2	2			1			2				
CO3	3	2	3	2	2	2			3		2				
CO4		2	3	3	3				2		3	3			
CO5		2	3	3	3		2		2		2	2			

2- Low Correlation; 2- Moderate Correlation; 3- Substantial Correlation

Integral University, Lucknow

Effective from Session: 2016-17							
Course Code	CS-302	**Title of the Course**	Designing and Analysis of Algorithm Lab	**L**	**T**	**P**	**C**
Year	III	**Semester**	V	0	0	2	1
Pre-Requisite	None	**Co-requisite**	None				
Course Objectives	• To learn the basic concepts of divide and conquers with help of various examples. • To be able to develop logics which help to find the optimal solution. • To learn the Dynamic approach to through various problems. • To learn the uses of augmented data structure and their implementation. • Learning Backtracking and its implementation.						

Course Outcomes	
CO1	Able to understand the basic concepts of Divide and conquer their implementation.
CO2	Able to understand and develop solution to optimization problem (Greedy algorithm)
CO3	Able to analyze and develop dynamic solution and implementation.
CO4	Develop understanding of Backtracking problems and their implementation.
CO5	Understanding and develop the logic to implementation of different augmenting data structures (RB Tree).

S. No.	**List of Experiments**	**Contact Hrs.**	**Mapped CO**
1	Implement Merge Sort.	2	1
2	Implement Quick Sort (Divide & Conquer)	2	1
3	Implement Heap Sort.	2	2
4	Implement Knapsack problem (Greedy ALGO.)	2	2
5	Implement of directed and undirected graph.	2	3
6	Implement Shortest path by Dijkstra Algorithm.	2	3
7	Implement 8- Queen problem (Back Tracking)	2	4
8	Implement Minimal spanning tree by • Kruskal's Algorithm • Prim's Algorithm	2	4
9	Implement Pattern Matching.	2	5
10	Implement Binary Search Tree.	2	5
11	Insert an element in Red Black Tree.	2	5

Reference Books:

1. Coremen, Rivest, Lisserson, "Algorithms", PHI.

2. Horwitz & Sahani, Fundamental of Computer Algorithm, Galgotia.

3. Michael T. Goodrich and Roberto Tamassia, Algorithm Design: Foundation, Analysis and Internet Examples, John Wiley Publications.

PO-PSO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	1	2	2		3	1	3	1					2	1	1
CO2	2	1	3	2	1	2	3					1	2	1	1
CO3	1	2	2	2			3	2				2	2	1	1
CO4		2	2	2	1	2	3	2					2	1	1
CO5	1	2	1		1		3					1	2	1	1

1-Low Correlation; 2- Moderate Correlation; 3- Substantial Correlation

Integral University, Lucknow

Effective from Session: 2016-17							
Course Code	CS-310	**Title of the Course**	OPEN SOURCE SOFTWARE TECHNOLOGIES LAB	**L**	**T**	**P**	**C**
Year	III	**Semester**	V	0	0	2	1
Pre-Requisite	None	**Co-requisite**	None				
Course Objectives	• To motivate students to use open source operating systems. • To teach students to setup their own Linux server. • To teach students to setup their own web server and commands • To learn using MySql as an open source database system. • To learn PHP as open source development programming language.						

Course Outcomes	
CO1	Explain common open source licenses and the impact of choosing a license to explain open- source project structure and how to successfully set up a project
CO2	Competent with Linux in their systems Install different useful packages in Linux using RPM can Schedule task automatically and run administrative commands.
CO3	Able to understand web server easily how to store, process and deliver web pages to the users. How intercommunication is done using by variety of available Protocols.
CO4	Analyze the existing design of a database schema and apply concepts of normalization to design an optimal database by formulating complex queries in MySQL.
CO5	Design and develop Client Server applications using open source scripting language. Able how to design GUI Applications in open source scripting language to evaluate different processes.

S. No.	**List of Experiments**	**Contact Hrs.**	**Mapped CO**
1	Overview of FOSS & Basic Command interface on Linux	2	1
2	Usage of Basic Linux Commands, File and Folder Management Commands	2	1
3	Learning Network related Command and Administrative Commands	2	1
4	Learning Vi Editor & its Modes and GUI Tools	2	2
5	Learning Shell Script, A Shell Script to demonstrate various control Constructs	2	2
6	A Script to check for a file and directory existence in the file system	2	2
7	A Script to execute different command to demonstrate Switch cases statement	2	3
8	A Script to handle command line argument and other Special symbols	2	3
9	Learn how to Compile, Debug & Execute C, C++ & Java Programming Codes without IDEs.	2	3
10	Learning about LAMP STACK its Installation And Configuration on Linux (Ubuntu) and Perform Post Installation Exercises	2	4
11	Creating simple Database in MySql Server performing queries	2	4
12	Learning A Deep Dive in MySql Server Using PhpMyAdmin Tool for Administering and Monitoring the Database Server, Mysql Admin, Backup and restore, User Account Rights Management	2	4
13	Basics of PHP Web Programming, PHP code to demonstrate the usage of Variable, String, Array and Control Structure	2	4
14	Some Deep Dive in PHP Programming: - A PHP Program to implement customized functions and other Form Handling Strategies	2	5
15	A PHP Program to demonstrate the use of PHP mail () function	2	5
16	Learning Database Connectivity between PHP and MySql, create a login Control for a web page to demonstrate the use of Connectivity and Basic retrieval of data from database	2	5
17	A Mini Project to create a website for University Utilities	2	5

PO-PSO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	2	3	1					2			3	3	2	
CO2	2	3	3	2					2			1	2	2	
CO3	3	3	2	2		2	3		2			1	1	2	
CO4	2	2	3	2	1	2			2			1	2	1	
CO5	2	2	2	2					2			1			3

1-Low Correlation; 2- Moderate Correlation; 3- Substantial Correlation

Integral University, Lucknow

Effective from Session: 2021-22							
Course Code	CS-393	**Title of the Course**	Hadoop Lab	**L**	**T**	**P**	**C**
Year	III	**Semester**	V	0	0	2	1
Pre-Requisite	None	**Co-requisite**	None				
Course Objectives	To provide an overview of an exciting field of big data analytics and Hadoop. Develop an understanding of the complete open-source Hadoop ecosystem and its near-term future direction. To introduce the tools required to manage and analyze big data like Hadoop, MapReduce To teach the fundamental techniques and principles in achieving big data analytics with scalability and streaming capability. To enable students to have skills that will help them to solve complex real-world problems in decision support.						

Course Outcomes	
CO1	To provide an overview of an exciting field of big data analytics and Hadoop
CO2	Develop an understanding of the complete open-source Hadoop ecosystem and its near-term future direction.
CO3	To introduce the tools required to manage and analyze big data like Hadoop, MapReduce
CO4	To teach the fundamental techniques and principles in achieving big data analytics with scalability and streaming capability.
CO5	To enable students to have skills that will help them to solve complex real-world problems in decision support.

S. No.	**List of Experiments**	**Contact Hrs.**	**Mapped CO**
1	Implement the following file management tasks in Hadoop:	2	1
2	• Adding files and directories • Retrieving files • Deleting files • Hint: A typical Hadoop workflow creates data files (such as log files) elsewhere and copies them into HDFS using one of the above command line utilities.	2	1
3	Install and Run Hive then use Hive to create, load, alter, and drop databases, tables.	2	2
4	Implement Hive Partitioning with data set	2	2
5	Implement Hive bucketing with data set.	2	3
6	Implement sqoop commands	2	3
7	Run a basic Word Count Map Reduce program to understand Map Reduce paradigm with data set.	2	4
8	Implement Hbase commands with data set	2	4
9	Install and Run Pig then write Pig Latin scripts to sort, group, join and filter your data	2	5
10	Explore Zookeeper	2	5
11	Explore Ambari	2	5

Reference Books:

- IBM Courseware
- IBM Courseware
- Predictive Analytics Mesmerizing & fascinating by ERIC SIEGEL

PO-PSO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1		2				3	2				2	2			
CO2		2	3	2				1	2		2				
CO3	3	2	3	2	2				3		2				
CO4		2	3	3	3		2		2		3	3			
CO5		2	3	3	2		2		2		3	2			

1-Low Correlation; 2- Moderate Correlation; 3- Substantial Correlation