Dr. Ambedkar Institute of Technology, Bengluru-56

(An Autonomous Institute, Approved by AICTE, Affiliated to V T U, Belagavi)
Nationally Accredited by NAAC with 'A' Grade

Master of Computer Applications

(Accredited by National Board of Accreditation)

MCA I to II Semester

(2018-21) Scheme & Syllabus

Dr. Ambedkar Institute of Technology, Bengaluru – 560056

Master of Computer Applications

Institute - Vision and Mission

Vision

To create Dynamic, Resourceful, Adept and Innovative Technical professionals to meet global challenges.

Mission

- To offer state-of-the-art under graduate, post graduate and doctoral programs in the fields of Engineering, Technology and Management
- To generate new knowledge by engaging faculty and students in research, development and innovation.
- To provide strong theoretical foundation to the students, supported by extensive practical training to meet the industrial requirements.
- To instill moral and ethical values with social and professional commitment.

DEPARTMENT - VISION AND MISSION

VISION

To create a quality human resource with good technical competence to face the global challenges.

MISSION

- To create a Center of Excellence through industry institute interaction.
- To prepare students for utilizing more creativity, innovativeness and leadership Qualities.
- To inculcate a sense of commitment to the students towards socio-economic development of the nation

Program Educational Objectives

PEO 1: Graduates are prepared to be employed in IT industries and be engaged in learning, understanding, and applying new ideas.

PEO 2: To prepare students for utilizing more creativity, innovativeness and leadership Qualities.

PEO 3: To provide an opportunity to students to learn the latest trends in computer technology and make them ready for life-long learning process with leadership skills and corporate social responsibilities.

Program Outcomes:

PO 1: Ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modelling and design of computer based systems.

PO 2: Graduates will be able to demonstrate with excellent programming, analytical, logical and problem solving skills.

PO 3: Graduates will be able to design a computing system to meet desired needs within realistic constraints such as safety, security and applicability.

PO 4: An ability to devise and conduct experiments, interpret data and provide well informed conclusions.

PO 5: An ability to select modern computing tools and techniques and use them with dexterity.

PO 6: An ability to function professionally with ethical responsibility as an individual as well as in multidisciplinary teams with positive attitude.

PO 7: An ability to appreciate the importance of goal setting and to recognize the need for life-long learning.

PO 8: Develop and maintain medium to large scale application software using theoretical and applied knowledge of software engineering and project management.

PO 9: An ability to communicate effectively.

PO 10: An ability to understand the impact of system solutions in a contemporary, global, economic, environmental, and societal context for sustainable development.

PO 11: An ability to execute the project either individually or in a group.

PO 12: An ability to become an Entrepreneur in the field of information technology to create a value and wealth for the betterment of the individual and society at large.

		I Semester							
Subject Code	**Name of the Subject**	**Credits**				**Exam Duration in Hours**	**Marks**		**Total Marks**
		Lecture	**Tutorial**	**Practicals**	**Total Credits**		**CIE**	**SEE**	
18MCA11	OOPS using C++	4	-	-	4	3	50	50	100
18MCA12	Linux Programming	4	-	-	4	3	50	50	100
18MCA13	Web Technologies	4	-	-	4	3	50	50	100
18MCA14	Discrete Mathematical Structures	4	-	-	4	3	50	50	100
18MCA15	Computer Organization and Architecture	3	-	-	3	3	50	50	100
18MCAL16	OOPS Lab	--	-	1	1	3	50	50	100
18MCAL17	Linux Lab	--	-	1	1	3	50	50	100
18MCAL18	Web Technology Lab	--	-	1	1	3	50	50	100
	Total	**19**	**-**	**3**	**22**	**24**	**400**	**400**	**800**

		II Semester							
Subject Code	**Name of the Subject**	**Credits**				**Exam Duration in Hours**	**Marks**		**Total Marks**
		Lecture	**Tutorial**	**Practicals**	**Total Credits**		**CIE**	**SEE**	
18MCA21	Java Programming	4	-	-	4	3	50	50	100
18MCA22	Data Structures using C++	4	-	-	4	3	50	50	100
18MCA23	Analysis and Design of Algorithms	4	-	-	4	3	50	50	100
18MCA24	Database Management System	4	-	-	4	3	50	50	100
18MCA25	Operating System	3	-	-	3	3	50	50	100
18MCAL26	Java Programming Lab	--	-	1	1	3	50	50	100
18MCAL27	Data Structures Lab	--	-	1	1	3	50	50	100
18MCAL28	DBMS lab	--	-	1	1	3	50	50	100
Total		**19**	**-**	**3**	**22**	**24**	**400**	**400**	**800**

		III Semester							
Subject Code	**Name of the Subject**	**Credits**				**Exam Duration in Hours**	**Marks**		**Total Marks**
		Lecture	**Tutorial**	**Practicals**	**Total Credits**		**CIE**	**SEE**	
18MCA31	Network Architecture & Management	4	-	-	4	3	50	50	100
18MCA32	Python Programming	4	-	-	4	3	50	50	100
18MCA33	Advanced Software Engineering	4	-	-	4	3	50	50	100
18MCA34	Data Science using R	3	-	-	3	3	50	50	100
18MCA35	Elective-1	3	-	-	3	3	50	50	100
18MCAL36	Network Architecture Lab	--	-	1	1	3	50	50	100
18MCAL37	Python programming Lab	--	-	1	1	3	50	50	100
18MCAL38	Data Science using R Lab	--	-	1	1	3	50	50	100
Total		**18**	**-**	**3**	**21**	**24**	**400**	**400**	**800**

ELECTIVE-1									
Subject Code	**Name of the Subject**	**Credits**				**Exam Duration**	**Marks**		**Total Marks**
		L	**T**	**P**	**Credits**		**CIE**	**SEE**	
18MCA351	Software Testing and Practices	3	-	-	3	3	50	50	100
18MCA352	Advanced DBMS	3	-	-	3	3	50	50	100
18MCA353	Artificial Intelligence	3	-	-	3	3	50	50	100

		IV Semester							
Subject Code	**Name of the Subject**	**Credits**				**Exam Duration in Hours**	**Marks**		**Total Marks**
		Lecture	**Tutorial**	**practicals**	**Total Credits**		**CIE**	**SEE**	
18MCA41	Enterprise Applications-1	3	-	-	3	3	50	50	100
18MCA42	Advanced Web Technologies	3	-	-	3	3	50	50	100
18MCA43	Analytical Skills And Building Professional Development	2	1	-	3	3	50	50	100
18MCA44	Elective-2	3	-	-	3	3	50	50	100
18MCA45	Elective-3	3	-	-	3	3	50	50	100
18MCA46	Research Methodology	2	-	--	2	3	50	50	100
18MCAL47	Enterprise Applications Lab	--	-	1	1	3	50	50	100
18MCAL48	Advanced Web Technologies Lab	--	-	1	1	3	50	50	100
18MCAM49	Mini Project using Android	--	1	1	2	3	50	50	100
Total		**16**	**2**	**3**	**21**	**27**	**450**	**450**	**900**

ELECTIVE-2									
Subject Code	**Name of the Subject**	**Credits**				**Exam Duration**	**Marks**		**Total Marks**
		L	**T**	**P**	**Credits**		**CIE**	**SEE**	
18MCA441	Information Security	3	--	--	3	3	50	50	100
18MCA442	Data Mining & Business Intelligence	3	--	--	3	3	50	50	100
18MCA443	Object oriented Modelling & Design	3	--	--	3	3	50	50	100

ELECTIVE-3									
Subject Code	**Name of the Subject**	**Credits**				**Exam Duration**	**Marks**		**Total Marks**
		L	**T**	**P**	**Credits**		**CIE**	**SEE**	
18MCA451	Software Quality & Performance Evaluation	3	--	--	3	3	50	50	100
18MCA452	Software Architecture	3	--	--	3	3	50	50	100
18MCA453	Enterprise Resource Planning	3	--	--	3	3	50	50	100

		V Semester							
Subject Code	**Name of the Subject**	**Credits**				**Exam Duration in Hours**	**Marks**		**Total Marks**
		Lecture	**Tutorial**	**practicals**	**Total Credits**		**CIE**	**SEE**	
18MCA51	Machine Learning	4	-	-	4	3	50	50	100
18MCA52	Big Data Analytics	3	-	-	3	3	50	50	100
18MCA53	Cloud Computing	3	-	-	3	3	50	50	100
18MCA54	Elective -4	3	-	-	3	3	50	50	100
18MCA55	Elective -5	3	-	-	3	3	50	50	100
18MCAL56	Machine Learning Lab	--	-	1	1	3	50	50	100
18MCAL57	Big Data Analytics Lab	--	-	1	1	3	50	50	100
18MCAL58	Cloud Computing Lab	--	-	1	1	3	50	50	100
18MCAI59	Industry Internship (6 weeks)	--	--	--	5	3	50	50	100
	Total	**16**	**-**	**3**	**24**	**27**	**450**	**450**	**900**

ELECTIVE-4									
V SEMESTER									
Subject Code	**Name of the Subject**	**Credits**				**Exam Duration**	**Marks**		**Total Marks**
		L	**T**	**P**	**Credits**		**CIE**	**SEE**	
18MCA541	Enterprise Application-2	3	--	--	3	3	50	50	100
18MCA542	Full Stack Development with MERN	3	--	--	3	3	50	50	100
18MCA543	DevOps	3	--	--	3	3	50	50	100

ELECTIVE-5									
V SEMESTER									
Subject Code	**Name of the Subject**	**Credits**				**Exam Duration**	**Marks**		**Total Marks**
		L	**T**	**P**	**Credits**		**CIE**	**SEE**	
18MCA551	Web Services	3	--	-	3	3	50	50	100
18MCA552	Internet of Things	3	--	-	3	3	50	50	100
18MCA553	Block Chain Technology	3	--	-	3	3	50	50	100

VI SEMESTER							
Subject Code	**Name of the Subject**	**Internal**	**External**	**Dissertation Evaluation**	**Viva voce**	**Total Marks**	**Credits**
18MCAS61	Technical Seminar	50	--	--	--	50	2
18MCAP62	Project Work	50	-	100	100	250	20
Total		100	--	100	100	300	22

Grand Total (I to VI Semesters): 4500 Marks : 132 Credits

SEMESTER –I Object Oriented Programming using C++			
Subject Code	**18MCA11**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**4**	**SEE Marks**	**50**
Total Number of Lecture Hours	**52**	**SEE Hours**	**03**
CREDITS – 4:0:0			

Course Learning Objectives(CLO)

- Develop a greater understanding of programming language and implementation
- Develop an in-depth understanding of functional, logic, and object-oriented programming paradigms
- Provide in-depth coverage of C and object-oriented programming concepts and techniques using C++.
- To demonstrate polymorphism of different operators which can be used based on the context of the program.
- To overcome the drawbacks of structured programming by introducing inheritance Concept which reduce the code complexity and increases the code reusability.
- Apply the standard template libraries for implementing stack, queue and list.

Modules

Module -1	**10 Hours**
Introduction to OOP with C++ Classes and Objects: The Origins of C++, What Is Object-Oriented Programming? ,Encapsulation, Polymorphism , Inheritance. The General Form of a C++ Program, Friend Functions, Friend Classes, Inline Functions, Constructors and Destructors, Constructors with One Parameter, When Constructors and Destructors Are Executed, Static Class Members ,Static Data Members ,Static Member Functions, The Scope Resolution Operator, Nested Classes, Passing Objects to Functions, Returning Objects, Object Assignment.	
Module -2	**10 Hours**
Arrays, Pointers, References, and the Dynamic Allocation Operators	

Arrays of Objects, Pointers, Pointer Expression, Pointer as function arguments, Functions returning pointers, Pointers to functions, Creating Initialized vs, Pointers to Objects, The this Pointer, References, Reference Parameters, Passing References to Objects, Returning References, C++'s Dynamic Allocation Operators new and delete, Initializing Allocated Memory ,Allocating Arrays ,Allocating Objects. Function Overloading, Copy Constructors and Default Arguments: Function Overloading, Overloading Constructors, Copy Constructors, Default Function Arguments, Function Overloading and Ambiguity. Operator Overloading: Creating a Member Operator Function, Creating Prefix and Postfix Forms, of the Increment and decrement Operators, Operator Overloading Restrictions, Operator Overloading Using a Friend Function, Using a Friend to Overload ++ or – –, Overloading [ ] , Overloading ()	
Module -3	**12 Hours**
Inheritance: Base-Class Access Control, Inheritance and protected Members, Protected Base-Class Inheritance, Inheriting Multiple Base Classes, Constructors, Destructors, and Inheritance, When Constructors and Destructors Are Executed, Passing Parameters to Base-Class Constructors, Granting Access, Virtual Base Classes. **Virtual Functions and Polymorphism:** Virtual Functions, Calling a Virtual Function Through a Base Class Reference, The Virtual Attribute Is Inherited, Virtual Functions are Hierarchical, Pure Virtual Functions, Abstract Classes, Using Virtual Functions, Early vs. Late Binding. Templates: Generic Functions, A Function with Two Generic Types, Explicitly Overloading a Generic Function, Overloading a Function Template, **Generic Function** Restrictions ,Applying Generic Functions, A Generic Sort, Generic Classes, Using Default Arguments with Template Classes, The type name and export Keywords, The Power of Templates.	
Module -4	**10 Hours**
Exception Handling: Exception Handling, Fundamentals, Catching Class Types, Using Multiple catch Statements, Handling Derived-Class Exceptions, Exception HandlingOptions, Catching All Exceptions, Restricting Exceptions, Rethrowing an Exception, Understanding terminate( ) and unexpected( ),uncaught_exception( ) Function, The exception and bad_exception Classes, Applying Exception Handling.	

Module -5	**10 Hours**
Standard C++ I/O Classes : Old vs. Modern C++ I/O, C++ Streams, The C++ Stream Classes, C++'s Predefined Streams, Formatted I/O, Formatting Using the ios Members, Setting the Format Flags, Clearing Format Flags, Overloading << and >>,Creating Your Own Inserters, Creating Your Own Extractors, Creating Your Own Manipulator Functions **C++ File I/O:** fstream> and the File Classes, Opening and Closing a File, Reading and Writing Text Files, put( ) and get( ), read( ) and write( ),More get( ) Functions, getline( ) , Detecting EOF, peek( ), Obtaining the Current FilePosition.	
Question paper pattern: • There will be 10 questions with 2 full questions from each module. • Each full question consists of 20 marks. • Students have to answer 5 full questions, selecting ONE from each module.	

Text Books:

1. ".Herbert Schildt: C++ The Complete Reference, 4th Edition, Tata McGraw Hill, 2014. (Listed topics only from Chapters 11,12,13,14, 15, 16, 17, 18, 19, 20, and21)
2. K R Venugopal, RajkumarBuyya,T Ravishanker: Mastering C++,Tata McGraw Hill.

Reference Books:

1. "Stanley B. Lippmann, JoseeLajore: C++Primer, 4th Edition, Addison Wesley.
2. Stephen Prata : C++ Primer Plus, 6th Edition, Pearson Education.
3. Object oriented programming with C++, E. Balaguruswamy, Tata McGraw Hill.

Course Outcome (CO):

CO1: Acquire knowledge on C++ programming concepts.
CO2: Analyze the different concepts of C++.
CO3: Design and Develop the solution to a problem using Object Oriented Programming Concepts.
CO4: Apply the learning into real world problems independently..

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO1,PO2
CO 2	PO1,PO2
CO 3	PO2,PO4
CO 4	PO2,PO4

CO 5	PO2,PO4

SEMESTER –I Linux Programming			
Subject Code	**18MCA12**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**4**	**SEE Marks**	**50**
Total Number of Lecture Hours	**52**	**SEE Hours**	**03**
CREDITS – 4:0:0			

Course Learning Objectives(CLO)

- Describe the architecture of Linix Operating System (OS).
- Demonstrate and analyse the Linix commands usage.
- Use Linix utilities to create simple tools for information processing.
- Illustrate the power of Linix shell by writing shell scripts.
- Explain and analyse the process concepts in Linix OS.

Modules	**Teaching Hours**
Module -1	**10 Hours**
Introduction History, Architecture, Experience the Basic commands ls cat,S, cal, date, calendar, who, printf, tty, sty, uname, passwd, echo, tput, bc, script, spell and spell, Command Line Arguments, Exit Status of a Command, The Logical Operator s&& and||,ex It ,if, and case conditions, expr, sleep and wait, while, until, for,$,@, redirection, set and shift, trap.	
Module -2	**10 Hours**
UNIX File System: The file, what's in a file name? The parent-child relationship, pwd the Home directory, Absolute path names, using absolute pathnames For a command, cd ,mkdir, rmdir, Relative path names, The UNIX file system. Basic File Attributes: Is–l, the-D option, File Permissions , chmod, Security and Security and File Permission,usersandgroups,security level,changing permission, Changing ownership and group, File Attributes, More file attributes 4hard link, symbolic link, umask, find.	
Module -3	**10 Hours**
Simple Filters: Pr, head, tail, cut, paste, sort , uniq, tr commands ,Filters using Filters using Regular Expression: grep, Regular Expression, egrep, fgrep sed instruction, Line Addressing, Insert ingand Changing Text, Context addressing, writing selected lines to a file, the– f option, Substitution ,Properties of Regular Expressions.	
Module -4	**10 Hours**

Awk and Advanced Shell Programming: Awk-Advanced Filters: Simple awk Filtering, Splitting a Line into Fields, printf, the Logical and Relational Operators, Number Processing, Variables, The –f option, BEGIN and END positional Parameters ,get line, Built-nvariables, Arrays, Functions, Interface with the Shell, Control Flow. The sh command export, the Command, Conditional Parameter substitution	
Module -5	**12 Hours**
Process and System Administration: Process basics, PS, internal and external commands, running jobs in background, nice, at and batch , cron, time commands, Essential system Administration root, administrator's privileges, startup & shutdown, managing disk space, cpio, tar, Customizing the Environment : System Variables, profile, sty, Aliases, Command History, On-line Command Editing.	

Question paper pattern:

- There will be 10 questions with 2 full questions from each module.
- Each full question consists of 20 marks.
- Students have to answer 5 full questions, selecting ONE from each module.

Text Books:

1. 1. Your UNIX-The Ultimate Guide, Sumitabha Das, Tata McGrawHill

Reference Books:

1. "Unix Shell Programming", Yashwant Kanetkar,
2. "Beginning Shell Scripting", Eric Foster-Johnson, JohnCWelch, MicahAnderson, Wroxpublication.
3. UNIX: Concepts and Applications, Sumitabha Das, Tata McGrawHill, "Introduction to UNIX" by M.G.Venkatesh Murthy.

Course Outcomes:

CO1: Understand and experience the UNIX environment, File system and hierarchy.
CO2: Understand File system and Demonstrate commands to extract, interpret data for further processing.
CO3: Understand Filters and Regular Expressions.
CO4: Analyze the usage of different shell commands, variables and AWK filtering.
CO5: Interpret and manipulate process attributes and System administrations

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO1, PO2
CO 2	PO1, PO2
CO 3	PO1, PO2

CO 4	PO1, PO2
CO 5	PO1

SEMESTER-I Web Technologies			
Subject Code	**18MCA13**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**4**	**SEE Marks**	**50**
Total Number of Lecture Hours	**52**	**SEE Hours**	**03**
CREDITS – 4:0:0			

Course Learning Objectives(CLO):

- Demonstrate the underlying principles, methods and approaches of Web technologies.
- Understand XHTML tags and CSS style sheets.
- Build Java script and different event handlers in java script.
- Demonstrate dynamic document using java script and XML.
- Use jQuery to develop dynamic and interactive web page.

Modules	Teaching Hours
Module -1	**9 Hours**
Web Fundamentals Internet, WWW, Web Browsers and Web Servers, URLs, MIME, HTTP, Security, and the Web Programmers Toolbox. Introduction to XHTML Basic syntax, Standard structure, Basic text markup, Images, Hypertext Links. Lists, Tables, Forms, Frames, syntactic differences between HTML and XHTML.	
Module -2	**12 Hours**
Introduction to HTML5 New features of HTML5, HTML5 DocType, HTML5 Structure, Tags- nav, section, article, aside, header, footer, HTML5 Form Elements- Search, tel, url, email, number and range, HTML5 Media tags- Audio and video. **Cascading Style Sheets** Introduction, Levels of style sheets, Style specification formats, Selector forms, Property value forms, Font properties, List properties, Color Alignment of text, The box model, Background images, The and tags, Conflict resolution.	
Module -3	**10 Hours**

The basics of JavaScript Overview of JavaScript, Object orientation and JavaScript, general Syntactic characteristics, Primitives, operations, and expressions, Screen output and keyboard input, Control statements, Object creation and modification, Arrays, Functions, Constructors, Pattern matching using regular expressions, Errors in scripts. **JavaScript and XHTML Documents** The JavaScript Execution Environment, The Document Object Model, Elements Access in Java Script, Events and Event Handling, Handling Events from Body Elements, Handling Events from Text Box and password Elements, The DOM2 Model, The navigator Object, Dom Tree Traversal and Modification.	
Module -4	**12 Hours**
Dynamic Documents with JavaScript Introduction, Positioning Elements, Moving Elements, Element Visibility, Changing Colors and Fonts, Dynamic Content, Stacking Elements, Locating the Mouse Cursor, Reacting to a Mouse Click, Slow Movement of Elements, Dragging and Dropping Elements **Introduction to XML Introduction** Syntax of XML, XML Document Structure, Document type definitions, Namespaces, XML schemas, displaying raw XML documents, Displaying XML documents with CSS, Web services.	
Module -5	**9 Hours**
Introduction to jQuery Introducing jQuery, jQuery fundamentals, Creating the wrapped element set, Bringing pages to life with jQuery, Understanding the browser event models, The jQuery Event Model, Sprucing up with animations and effects.	

Question paper pattern:

- There will be 10 questions with 2 full questions from each module.
- Each full question consists of 20 marks.
- Students have to answer 5 full questions, selecting ONE from each module.

Text Books:

1. Robert W.Sebesta ,"Programming the World Wide Web", 4thEdition, Pearson education, 2012.
2. Kogent Learning solutions Inc., "HTML 5: Covers CSS3, JavaScript,XML, XHTML AJAX, PHP & JQuery: Black Book", Dreamtech Press.

3. Bear Bibeault, Yehuda Katz: jQuery in Action. 3rd Edition, DreamTech India,2008.

Reference Books:

1. Randy Connolly, Ricardo Hoar, "Fundamentals of Web Development", 2nd Edition, Pearson, 2018.
2. Jeffrey C.Jackson: Web Technologies-A Computer Science Perspective, Pearson Education, 7 th Impression, 2012.
3. Chris Bates: Web Programming Building Internet Applications, 3rdEdition, WileyIndia, 2009.
4. Zak Ruvalcaba Anne Boehm, "Murach's HTML5 and CSS3", 3rd Edition, Murachs/Shroff Publishers & Distributors Pvt Ltd, 2016.

COURSE OUTCOMES:

CO1: Understand the fundamentals of web and outline the features.

CO2: Design a web page with media components using HTML5 and CSS.

CO3: Develop XML documents and display using CSS.

CO4: Design and develop dynamic and interactive web pages using JavaScript and jQuery.

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO5,PO11
CO 2	PO5,PO11
CO 3	PO4,PO5,PO7,PO11
CO 4	PO2,PO4,PO5,PO11

SEMESTER –I

DISCRETE MATHEMATICAL STRUCTURES

Subject Code	**18MCA14**	CIE Marks	**50**
Number of Lecture Hours/Week	**4**	SEE Marks	**50**
Total Number of Lecture Hours	**52**	SEE Hours	**03**

CREDITS – 4:0:0

Course Learning Objectives(CLO):

- Derive logical implications and equivalences using laws of logic, use of quantifiers and different ways of proving a given statement.
- Apply the concept of set theory and properties of relations, draw Hasse diagram for partial ordering and lattice.
- Apply axioms of probability and derive independent and dependent probability.
- Identify types of graph, outline properties of graphs, and apply to some practical problems like traveling sales man problem.

Modules	**Teaching Hours**
Module -1	**12 Hours**
Fundamentals of Logic: Basic connectives and truth tables, logical equivalence, laws of logic, logical Implication, Rules of inference, Use of Quantifiers.	
Module -2	
Sets Theory and Probability Sets and subsets,set operations, laws of set theory, counting and venn diagrams. A first word on probability, axioms of probability, conditional probability.	**10 Hours**
Module -3	**10 Hours**
Fundamentals of Counting and Properties of Integers: The rules of Sum and Product, Permutations and Combinations, The Binimial theorem, Mathematical Induction, Recursive definitions: Fibonacci and Lucas numbers	
Module -4 **Relations and Functions** Cartesian product and relations ,Properties of relations, Equivalence relations, Posets and Hasse diagrams.	**10 Hours**

Functions Functions: plain and one-to-one, onto functions, Stirling numbers of the second kind, Pigeon hole principle.	
Module -5	**10 Hours**
Graph Theory: Graphs, terminology and special types of graphs, representation of graphs, isomorphism, connectivity, Euler and Hamiltonian paths, shortest path problems	

Question paper pattern:

- There will be 10 questions with 2 full questions from each module.
- Each full question consists of 20 marks.
- Students have to answer 5 full questions, selecting ONE from each module.

Text Books:

1. Discrete and Combinatorial Mathematics by Ralph P. Grimaldi and B V Ramana,5thedition,Pearson,2011. (Chapters: 1.1 to 1.3 4.1, 4.2,2.1 to 2.5, 3.1 to 3.6)
2. Higher Engieering Mathematics by Dr. B. S. Grewal, Khanna publishers, 40th edition(Chapters: 25.12 to 25.16, 24.1 to 24.6)

Reference Books:

1. Graph Theory with Applications to Engineering and Computer Science by NarsinghDeo, Prentice Hall India, 2004.
2. Discrete Mathematical Structures with Applications to Computer Science by J. P. Tremblay and R. Manohar, McGraw Hill.

Course Outcome (CO):

CO1: Use the logical notation to define and reason about proofs and disproof's.

CO2: Apply fundamental mathematical concepts such as sets, relations, and functions.

CO3: Calculate numbers of possible out comes of elementary combinatorial processes such as permutations and combinations.

CO4: Apply graph theory models of data structures and state machines to solve problems of connectivity and constraint satisfaction.

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO1,PO2
CO 2	PO1,PO2
CO 3	PO1,PO2
CO 4	PO1,PO2
CO 5	PO1,PO2

SEMESTER –I

Computer Organization and Architecture

Subject Code	**18MCA15**	CIE Marks	**50**
Number of Lecture Hours/Week	**3**	SEE Marks	**50**
Total Number of Lecture Hours	**39**	SEE Hours	**03**

CREDITS – 3:0:0

Course Learning Objectives(CLO)

- To understand the design of Digital Logic System.
- To study the fundamental working of functional units of a computer
- To study the design of instructions in a basic system and interfaces
- To study the functionalities of hierarchical memory system

Modules	**Teaching Hours**
Module -1	**8 Hours**
Computer Organizations: -Binary Systems, Combinational Logic and Sequential Logic: Digital computers and digital systems, binary numbers, number base conversion octal and hexadecimal numbers, binary code, binary storage and registers binary logic, integrated circuits digital logic gates Map Method (2,3,4 variable Map) NAND and NOR implementation , don't care condition.	
Module -2	**8 Hours**
Introduction to Adders, Subtractors, Decoders, Multiplexers, Sequential logic: Introduction, Different types of Flip Flops, Registers.	
Module -3	**8 Hours**
Computer Types, Functional Units, Basic Operational Concepts, Bus structure, Software, Performance, Multiprocessing and Multi computers, Introduction to Assemblers and Compilers. Machine Instruction: Memory Locations and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes.	
Module -4	**8 Hours**
Basics of Assembly Language Programme, Examples from Assembly Language Programming. Accessing I/O Devices, Interrupts, DMA.	

Module -5	**7 Hours**
Basic Concepts, Semiconductor RAM Memories, Read – Only Memories, Speed, Size, and Cost, Cache Memories, Virtual Memories, Memory Management Requirements, Secondary Storage.	

Question paper pattern:

- There will be 10 questions with 2 full questions from each module.
- Each full question consists of 20 marks.
- Students have to answer 5 full questions, selecting ONE from each module.

Text Books:

1. M. Morris Mano, " Digital Logic and Computer Design", Pearson, 2012 (Chapters1.1 - 1.4, 1.6 - 1.9, 2.7, 3.2 - 3.3, 3.6, 3.8, 4.3 - 4.4, 5.5- 5.6, 6.1-6.2 7.2 - 7.4)
2. Carl Hamachar and Z. V. S. Zaky, " Computer Organization", 5th Edition, Tata McGraw Hill (Chapters 1.1 - 1.5, 2.1 - 2.4, 4.1 - 4.4, 5.1 - 5.5, 5.7)
3. P. V. S. Rao, " Computer System Organization and Architecture", Pearson 2009 (Chapters 8.1 - 8.7 , 10.1 – 10.5

Reference Books:

1. JohnP.Hayes, "Computer Architecture and Organization", Tata McGraw-Hill, Edition, 2012.
2. SoumitrsKumar Mandal,"Digital Electronics Principles and Applications", Tata McGraw-Hill, 2010
3. Hamacher , " Computer Organization" , McGraw-Hill Education

Course Outcome (CO):

CO1: Understand and apply the concepts in the design of a logic system
CO2:. Understand the Basics of Computer system organization
CO3: Analyse and implement the addressing modes and instruction set.
CO4: Acquire knowledge on I/O interfaces and memory hierarchy

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO1
CO 2	PO1,PO2
CO 3	PO2
CO 4	PO2
CO 5	PO2

SEMESTER –I

OOPS using C++ LAB

Laboratory Code	**18MCAL16**	CIE Marks	**50**
Number of Lecture Hours/Week	**2**	SEE Marks	**50**

SEE Hours :**03**

CREDITS – 0:0:1

Course Learning Objective(CLO)

CO1: To learn and implement how to solve common types of computing problems.
CO2: To acquire knowledge on concepts of C++ programming.
CO3: To improves the problem solving skills using C++.
CO4: To design and develop interactive applications using C++.
CO5: To implement real world problems using oop's.

PART A

1. Write a C++ program to find the sum for the given variables using function with default arguments.

2. Write a C++ program to swap the values of two variables and demonstrates a function using call by value.

3. Write a C++ program to swap the values of two variables and demonstrates a function using Call by reference using reference type (&).

4. Write a C++ program to swap the values of two variables and demonstrates a function using Call by reference using pointer (*).

5. Write a C++ program to swap the values of two dynamically allocated variables and release the memory after swapping. (use new & delete operators)

6. Write a program to find the largest, smallest & second largest of three numbers. (use inline function MAX and MIN to find largest & smallest of 2 numbers)

7. Write a program to calculate the volume of different geometric shapes like cube, cylinder and sphere and hence implement the concept of Function Overloading.

8. Write a C++ program to create a template function for Bubble Sort and demonstrate sorting of integers and doubles.

PART B

1. Define a STUDENT class with USN, Name, and Marks in 3 tests of a subject. Declare an array of 10 STUDENT objects. Using appropriate functions, find the average of the two better marks for each student. Print the USN, Name and the average marks of all the students.

2. Write a C++ program to create a class called COMPLEX and implement the following overloading functions ADD that return a complex number:
(i) ADD (a, s2) – where 'a' is an integer (real part) and s2 is a complex number
(ii) ADD (s1, s2) – where s1 and s2 are complex numbers.

3. Friend functions and friend classes:

a) Write a program to define class name HUSBAND and WIFE that holds the income respectively. Calculate and display the total income of a family using Friend function.

b) Write a program to accept the student detail such as name and 3 different marks by get_data() method and display the name and average of marks using display() method. Define a friend class for calculating the average of marks using the method mark_avg().

4. Create a class called MATRIX using two-dimensional array of integers. Implement the following operations by overloading the operator == which checks the compatibility of two matrices to be added and subtracted. Perform the addition and subtraction by overloading the + and – operators respectively. Display the results by overloading the operator <<. If (m1== m2) then m3 = m1+m2 and m4 = m1- m2 else display error.

5. Write a program to create an HUMAN class with features as number of Head, Legs, Hands.(NOTE: Number of Head, Legs and Hands are of integer types)

a. Create an object HUMAN1 using default constructor. (Default features to have 1 Head, 2 Legs and 2 Hands)

b. Create an object HUMAN2 with customized inputs using Parameterized Constructor

c. Create an object HUMAN3 using existing object HUMAN1 (Copy Constructor).

d. All Humans die after their lifetime. (Destructor)

6. Demonstrate Simple Inheritance concept by creating a base class FATHER with data members SurName and BankBalance and creating a derived class SON, which inherits SurName and BankBalance feature from base class but provides its own feature FirstName and DOB. Create and initialize F1 and S1 objects with appropriate constructors and display the Father & Son details. (Hint : While

creating S1 object, call Father base class parameterized constructor through derived class by sending values)

7. Create an abstract base class EMPLOYEE with data members: Name, EmpID and BasicSal and a pure virtual function Cal_Sal().Create two derived classes MANAGER (with data members: DA and HRA and SALESMAN (with data members: DA, HRA and TA). Write appropriate constructors and member functions to initialize the data, read and write the data and to calculate the net salary. The main() function should create array of base class pointers/references to invoke overridden functions and hence it implements run-time polymorphism.

8. Write a program to implement FILE I/O operations on characters. I/O operations includes inputting a string, Calculating length of the string, Storing the string in a file, fetching the stored characters from it, etc.

9. Write a program to implement Exception Handling with minimum 5 exceptions Classes including two built-in exceptions.

10. a.Write a program to concatenate 2 strings using STL String class functions.

b.Write a simple C++ program to store and display integer elements using STL Vector class.

Note 1: In the practical Examination, each student has to pick one question from PART A and one from Part B.

Note 2: Change of program is not permitted in the Practical Examination

Course Outcomes (CO):

CO1: Apply and implement major programming and object oriented concepts like function overloading, operator overloading, Encapsulations, and inheritance, message passing to solve real-world problems.
CO2: Use major C++ features such as Virtual functions, Templates for data type independent designs and File I/O to deal with large data sets.
CO3: Analyze, design and develop solutions to real-world problems applying OOP Concepts of C++.

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO1, PO2

SEMESTER –I

LINUX LAB

Laboratory Code	**18MCAL17**	CIE Marks	**50**
Number of Lecture Hours/Week	**02**	SEE Marks	**50**

SEE Hours : **03**

CREDITS – 0:0:1

Course Learning Objectives(CLO) :

- Familiarize the Unix environment
- Learn to work on Vi-editor
- Understand basic commands
- Apply commands and to write the shell scripts
- Understand the Filters and to use the basic filters

A. Explore the UNIX environment.

B. Explore vi editor with vim tutor .Perform the following operations using vi editor, but not limited to:

1. Insert character, delete character, replace character
2. Save the file and continue working
3. save the file and exit the editor
4. quit the editor
5. quit without saving the file
6. rename a file
7. insert lines, delete lines,
8. setline numbers
9. search for a pattern
10. move forward and backward

1a. Write a shell script that takes a valid directory name as an argument and recursively descend all the sub-directories, finds the maximum length of any file in that hierarchy and writes this maximum value to the standard output.

1b. Write a shell script that accepts a pathname and creates all the components in that pathname as directories. For example, if the script is named mpc,then the command mpc a/b/c/d should create directories a, a/b, a/b/c, a/b/c/d.

2a.Write a shell script that accepts two filenames as arguments, checks if the permissions for these files are identical and if the permissions are identical, output

common permissions and otherwise output each filename followed by its permissions.

2b.Write a shell script which accepts valid log-in names as arguments and prints their corresponding home directories, if no arguments are specified, print a suitable error message.

3a. Create a script file called file-properties that reads a filename entered and outputs it Properties

3b. Write shell script to implement terminal locking (similar to the lock command).It should prompt the user for a password. After accepting the password entered by the user, it must prompt again for the matching password as confirmation and if match occurs, it mus lock the keyword until a matchingpassword is entered again by the user, Note that the script must be written to disregard BREAK, control-D. No time limit need be implemented for the lock duration.

4a. Write a shell script that accept one or more file names as argument and convert all of them to uppercase, provided they exist in current directory.

4b. Write a shell script that displays all the links to a file specified as the first argument to the script. The second argument, which is optional, can be used to specify in which the search is to begin. If this second argument is not present, the search is to begin in current working directory. In either case, the starting directory as well as all its subdirectories at all levels must be searched. The script need not include any error checking.

5a. Write a shell script that accepts as filename as argument and display its creation time if file exist and if it does not send output error message.

5b. Write a shell script to display the calendar for current month with current date replaced by *or** depending on whether the date has one digit or two digits

6a. Write a shell script to find a file/s that matches a pattern given as command line argument in the home directory, display the contents of the file and copy the file into the directory ~/mydir.

6b. Write a shell script to list all the files in a directory whose filename is atleast 10 characters.(use expr command to check the length)

7a. Write a shell script that gets executed displays the message either “Good Morning” or “Good Afternoon” or “Good Evening” depending upon time at which the user logs in.

7b. Write a shell script that accept a list of filenames as its argument, count and report occurrence of each word that is present in the first argument file on other

argument files.

8a. Write a shell script that determine the period for which a specified user is working on system and display appropriate message.

8b. Write as hell script that reports the logging in of a specified user with in one minute after he/she login. The script automatically terminate if specified user does not login during a specified period of time

9a. Write a shell script that accept the filename, starting and ending line number as an argument and display all the lines between the given line number.

9b. Write a shell script that folds long lines into 40columns.Thus any line that exceeds 40characters must be broken after 40^{th},a"\" is to be appended as the indication of folding and the processing is to be continued with the residue. The input is to be supplied through at ext file created by the user.

10a. Write an awk script that accepts date argument in the form of dd-mm-yy and displays it in the form if month, day and year. The script should check the validity of the argument and in the case of error, display a suitable message.

10b. Write an awk script to delete duplicated line from a text file. The order of the original lines must remain unchanged.

11a. Write an awk script to find out total number of books sold in each discipline as well as total book sold using associate array down table as given below.

Electrical	34
Mechanical	67
Electrical	80
ComputerScience	43
Mechanical	65
Civil	98
ComputerScience	64

11b. Write an awk script to compute gross salary of an employee accordingly to rule given below.

If basicsalary is<10000 then HRA=15% of basic & DA=45% of basic If basicsalary is >=10000 then HRA=20% of basic & DA=50% of basic.

Note 1: In the practical Examination each student has to pick one question from a lot of all1-11 questions.

COURSE OUTCOMES(CO):

CO1: Understand the Unix programming environment.

CO2: Be fluent in the use of Vi editor.

CO3: Be able to design and implement shell scripts to manage users with different types of Permission and file based applications.
CO4: Be fluent to write Awk scripts.

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO1,PO2,PO3,PO4,PO5,PO7,PO8,PO11

SEMESTER –I

Web Technologies Lab

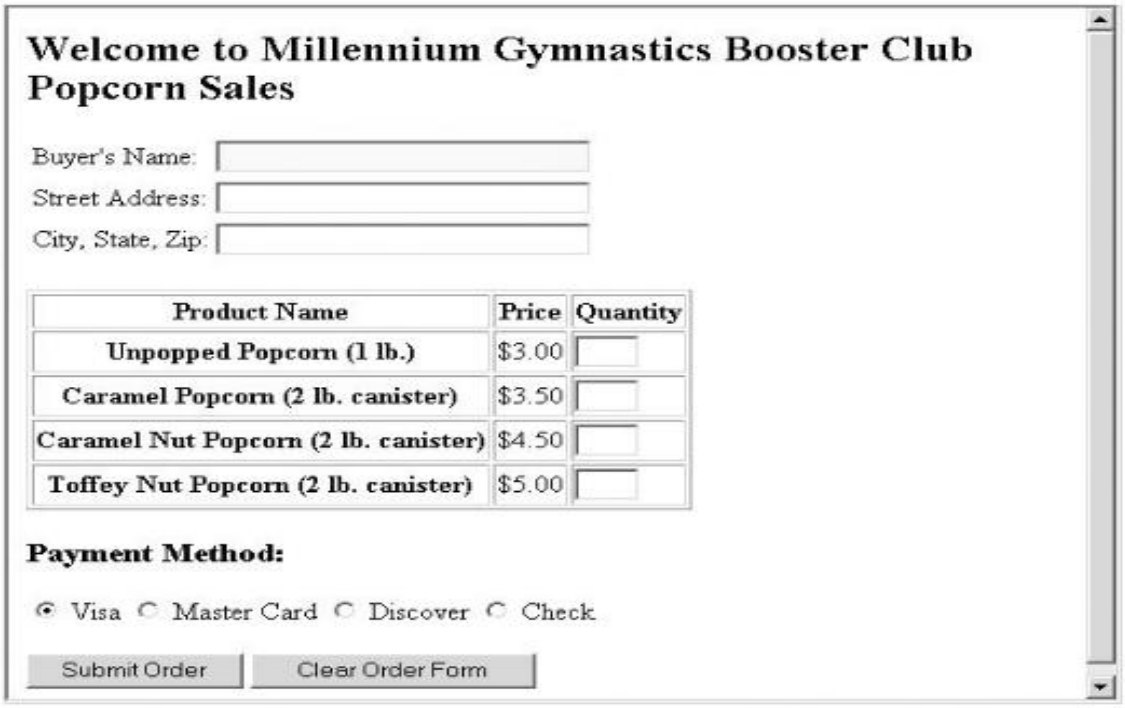
Laboratory Code	**18MCAL18**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**02**	**SEE Marks**	**50**
		SEE Hours	**03**

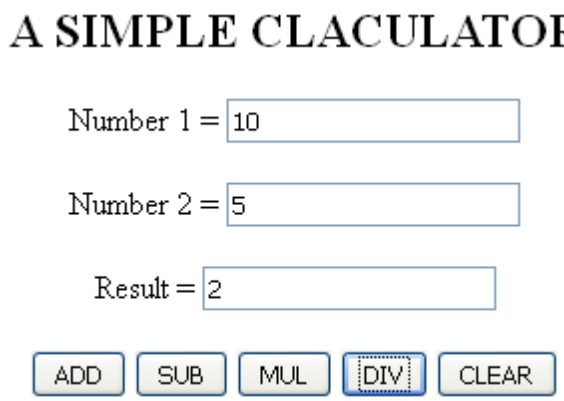
CREDITS – 0:0:1

Course Learning Objectives(CLO):

- To develop web pages using HTML and HTML5.
- To demonstrate the usage of CSS in designing web pages.
- To execute simple programming questions using JavaScript.
- To create dynamic web pages by manipulating the DOM elements.
- To design and implement user interactive dynamic web based applications using jQuery.

1	Create an XHTML page that provides information about your department. Your XHTML page must use the following tags: a) Text Formatting tags b) Horizontal rule c) Meta element d) Links e) Images f) Tables (Use of additional tags encouraged).
2	Develop and demonstrate the usage of inline, external and internal style sheet using CSS. Use XHTML page that contains at least three paragraphs of text, listed elements and a table with four rows and four columns.
3	Develop and demonstrate a XHTML file that includes Javascript script for the following problems: a) Input : A number n obtained using prompt Output : The first n Fibonacci numbers b) Input : A number n obtained using prompt Output : A table of numbers from 1 to n and their squares using alert
4	Write a JavaScript program to generate n number of random numbers and store them in an array. Sort the generated numbers in ascending order using array sort method. Develop separate functions to find mean and median of numbers that are in the array. Display the results with appropriate messages.
5	Create a XHTML document that describes the form for taking orders for popcorn. Text boxes are used at the top of the form to collect the buyer's name and address. These are placed in a borderless table to force the text box align vertically. A second table to collect actual order. Each row of this table names a product, displays the price, and uses text box with size 2 to collect the quantity ordered using <td> tag. The payment method is input by the user through one of four radio buttons. Provide provision for submission of order and clear the order form. **Sample Output**

	
6	Develop, test and validate an XHTML document that has checkboxes for apple (59 cents each), orange (49 cents each), and banana (39 cents each) along with submit button. Each checkboxes should have its own onclick event handler. These handlers must add the cost of their fruit to a total cost. An event handler for the submit button must produce an alert window with the message 'your total cost is $xxx', where xxx is the total cost of the chose fruit, including 5 percent sales tax. This handler must return 'false' (to avoid actual submission of the form data). Modify the document to accept quantity for each item using textboxes.
7	a) Develop and demonstrate, a HTML document that collects the USN (the valid format is : A digit from 1 to 4 followed by two upper-case characters followed by two digits followed by three upper-case characters followed by two digits; (no embedded spaces are allowed) from the user. Use JavaScript that validate the content of the document. Suitable messages should be display in the alert if errors are detected in the input data. Use CSS and event handlers to make your document appealing. b)Modify the above program to get the current semester also(restricted to be a number from 1 to 6)
8	Develop and demonstrate a HTML5 page which contains a) Dynamic Progressive bar. b) Display Video file using HTML5 video tag.
9	Develop and demonstrate, using JavaScript script, a XHTML document that contains three short paragraphs of text, stacked on top of each other, with only enough of each showing so that the mouse cursor can be placed over some part of them. When the cursor is placed over the exposed part of any paragraph, it

	should rise to the top to become completely visible. Modify the above document so that when a text is movedfrom the top stacking position, it returns to its original position rather than to the bottom
10	Develop a simple calculator to perform arithmetic (addition, subtraction, multiplication and division) operations on given two numbers. Use an html tag that allows the user to input two numbers and to display the result of arithmetic operation. Write suitable HTML and JavaScript and CSS to your simple calculator. The following figure show sample document display. Modify your program to make HTLM document as eye-catching using CSS.. 
11	Design an XML document to store information about a student in an engineering college affiliated to VTU. The information must include USN, Name, and Name of the College, Branch, Year of Joining, and e-mail id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.
12	Develop and demonstrate using jQuery to solve the following: a) Limit character input in the text area including count. b) Based on check box, disable/enable the form submit button.

Note 1: In the practical Examination each student has to pick one question from a lot of all 12 Questions

Course Outcomes(CO):

CO1: Design and implement user interactive dynamic web based applications using XHTML5,CSS, JAVA SCRIPT,XML & jquery

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO1,PO2,PO4,PO5,PO11

Dr. Ambedkar Institute of Technology, Bengaluru-56

(An Autonomous Institute, Affiliated to VTU, Belagavi)

Master of Computer Applications Program

(Accredited by National Board of Accreditation)

MCA II semester Syllabus

(2018 – 2021)

SEMESTER –II

JAVA PROGRAMMING

Subject Code	**18MCA21**	CIE Marks	**50**
Number of Lecture Hours/Week	**4**	SEE Marks	**50**
Total Number of Lecture Hours	**52**	SEE Hours	**03**

CREDITS – 4:0:0

Course Learning Objectives(CLO):

- Understand the different object oriented concepts and implement basic programs.
- Develop applications using inheritance and interface concepts.
- Apply multithreading programming concepts and handling errors efficiently.
- Design client server application in java

Modules	**Teaching Hours**
Module -1	**11 Hours**
Java Programming Fundamentals The Java Language, The Key Attributes of Object-Oriented Programming, The Java Development Kit, A First Simple Program, Handling Syntax Errors, The Java Keywords, Identifies in Java, The Java Class Libraries. **Introducing Data Types and Operators** Java's Primitive Types, Literals, A Closer Look at Variables, The Scope and Lifetime of Variables, operators, Shorthand Assignments, Type conversion in Assignments, Using Cast, Operator Precedence, Expressions. **Program Control Statements** Input characters from the Keyword, if statement, Nested ifs, if-else-if Ladder, Switch Statement, Nested switch statements, for Loop, Enhanced for Loop, While Loop, do-while Loop, Use break, Use continue, Nested Loops**.** **More Data Types and Operators**	

Arrays, Multidimensional Arrays, Alternative Array Declaration Syntax, Assigning Array References, Using the Length Member, The For-Each Style for Loop, Strings, The Bitwise operators. **String Handling** String Fundamentals, The String Constructors, Three String-Related Language Features, The Length () Method, Obtaining the characters within a string, String comparison, using indexOf() and lastIndexOf(), Changing the case of characters within a string, String Buffer and String Builder.	
Module -2	**11 Hours**
Introducing Classes, Objects and Methods Class Fundamentals, How Objects are Created, Reference Variables and Assignment, Methods, Returning from a Method, Returning Value, Using Parameters, Constructors, Parameterized Constructors, The new operator Revisited, Garbage Collection and Finalizers, The this Keyword. **A Closer Look at Methods and Classes** Controlling Access to Class Members, Pass Objects to Methods, How Arguments are passed, Returning Objects, Method Overloading, Overloading Constructors, Recursion, Understanding Static, Introducing Nested and Inner Classes, Varargs: Variable-Length Arguments. **Inheritance** Inheritance Basics, Member Access and Inheritance, Constructors and Inheritance, Using super to Call Superclass constructors, Using super to Access Superclass Members, Creating a Multilevel Hierarchy, When are Constructors Executed, Superclass References and Subclass Objects, Method Overriding, Overridden Methods support polymorphism, Why Overridden Methods, Using Abstract Classes, Using final, The Object Class.	
Module -3	**10 Hours**
Interfaces Interface Fundamentals, Creating an Interface, Implementing an Interface, Using Interface References, Implementing Multiple Interfaces, Constants in Interfaces, Interfaces can be extended, Nested Interfaces, Final Thoughts on Interfaces. **Packages**	

Package Fundamentals, Packages and Member Access, Importing Packages, Static Import **Exception Handling** The Exception Hierarchy, Exception Handling Fundamentals, The Consequences of an Uncaught Exception, Exceptions Enable you to handle errors gracefully, using Multiple catch clauses, Catching subclass Exceptions, try blocks can be nested, Throwing an Exception, A Closer look at Throwable, using finally, using throws, Java's Built-in Exceptions, New Exception features added by JDK 7, Creating Exception Subclasses.	
Module -4	**10 Hours**
Multithreaded Programming Multithreading fundamentals, The Thread Class and Runnable Interface, Creating Thread, Creating Multiple Threads, Determining When a Thread Ends, Thread Priorities, Synchronization, using Synchronization Methods,The Synchronized Statement, Thread Communication using notify(), wait() and notify All(), suspending, Resuming and stopping Threads. **Enumerations, Auto boxing and Annotations** Enumerations, Java Enumeration are class types, The Values () and Valueof() Methods, Constructors, methods, instance variables and enumerations, Auto boxing, Annotations (metadata), **Generics** **Applets** Applet basics, A complete Applet Skeleton, Applet Initialization and Termination, A key Aspect of an Applet Architecture, Requesting Repainting, using the status window, Passing parameters to Applets.	
Module -5	**10 Hours**
Networking with Java.net Networking fundamentals, The Networking classes and Interfaces, The InetAddress class, The Socket Class, The URL class, The URLConnection Class, The HttpURL Connection Class. **Exploring Collection Framework** Collections Overview, The Collection Interfaces, The collection Classes. The Arrays Class.	
Question paper pattern: • There will be 10 questions with 2 full questions from each module.	

- Each full question consists of 20 marks.
- Students have to answer 5 full questions, selecting ONE from each module.

Text Books:

1. Java Fundamentals, A comprehensive Introduction by Herbert Schildt, Dale Skrien. Tata McGraw Hill Edition 2013.

Reference Books:

1. Java Programming by Hari Mohan Pandey, Pearson Education, 2012.

2. Java 6 Programming, Black Book, KoGenT ,Dreamtech Press, 2012.

3. Java 2 Essentials, Cay Hortsmann, second edition, Wiley

Course Outcomes(CO):

CO1: Demonstrate the basic object oriented concepts & apply them to create java applications.

CO2: Apply inheritance and interface concepts to design java applications.

CO3: Design java applications with multithreading concepts and demonstrate the error handling concepts.

CO4: Design client server applications.

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO1,PO2,PO3,PO4,PO5,PO7,PO12
CO 2	PO1,PO2,PO3,PO5,PO7,PO9
CO 3	PO1,PO2,PO3,PO5,PO7,PO9
CO 4	PO1,PO2,PO4,PO5,PO7,

SEMESTER –II

DATA STRUCTURE USING C++

Subject Code	**18MCA22**	CIE Marks	**50**
Number of Lecture Hours/Week	**4**	SEE Marks	**50**
Total Number of Lecture Hours	**52**	SEE Hours	**03**

CREDITS – 4:0:0

Course Learning Objectives(CLO):

- Acquire knowledge on Primitive and Non- Primitive data types and ADT
- Analyze different types of data structures
- Demonstrate different Applications of data structures.
- Demonstrate the usage and implementation of different type of Trees

Modules	**Teaching Hours**
Module -1	**10 Hours**
Introduction to Data Structures: **Abstract Data Type:** Definition, Abstract Data Type, Model for an ADT, ADT Implementations and example. **Recursion:** Recursive definition and designing recursive algorithms, Example on recursion: GCD, Fibonacci numbers, Towers of Hanoi problem. **Stack and its applications:** Definition and examples, Representing stacks, Example – Infix, Prefix and Postfix, Evaluation of Postfix Expression.	
Module -2	**12 Hours**
Queues: Queue operations, Queue ADT. Queue Applications. **Lists:** Basic Operations, Implementation, List ADT, Applications, Complex Implementations.**Singly Linked List**. Linked list as a data Structure, Inserting and removing nodes from a list, Linked implementations of stacks, Example of list operations such as insert and delete an element before a key element, Header nodes, Array implementation of lists.: **Circular Linked List**: Inserting, deleting and searching elements in a lists, **Double Linked List**: Inserting and	

Deleting Nodes, Queue as doubly linked lists, such as insert into position, Delete an specified element	
Module -3	**10 Hours**
Trees: Trees: Basic concepts, Binary trees and its properties, operations on binary trees, Binary tree traversals, Expression tree, General trees: insertions, deletions, changing a general tree to a binary tree Binary Search Tree: Binary Search tree: Basic concepts and its operations, threaded trees.	
Module 4	**10 Hours**
Advanced concepts in Trees: AVL Search trees: Basic concepts, implementations, Heaps - Basic Concepts, implementation**.**	
Module -5	**10 Hours**
Multiway Trees: M-way search trees, B-trees: Basic concepts, Implementations, Simplified B-Trees: 2-3 tree, 2-3-4 tree	

Question paper pattern:

- There will be 10 questions with 2 full questions from each module.
- Each full question consists of 20 marks.
- Students have to answer 5 full questions, selecting ONE from each module.

Text Books:

1.Richard F Gilberg and BehrouzAForouzan: Data Structures - A Pseudocode Approach with C,Cengage Learning, 6 the Indian Reprint 2009. Chapters (1.2, 1.3, 1.4, 2.2, 2.3, 3.1, 3.2, 3.3 3.4, 3.5, 4.1, 4.2, 4.3, 4.5, chapter 5, 6.1, 6.2, 6.3, 7.1, 7.2, 7.3, 7.5, 8.1, 8.2, 9.1, 9.2, 10.1, 10.2, 10.4, 11.1, 11.2, 11.3, 11.4) 30

Reference Books:

1. YedidyahLangsam and Moshe J. Augenstein and Aaron M Tenenbaum: Data Structures using C and C++, 2ndEdition, Pearson Education Asia, 2002.

2. NanjeshBennur, Dr.Manjaiaha DH, Dr. C.K. Subbaraya: C programming skills and Data Structures primer, First Edition, IPH Publication, 2017.

3. Mark Allen Weiss: Data Structures and Algorithm Analysis in C, 2nd Edition, Pearson Education Asia, 1997.

Course outcomes(CO):

CO1: Apply the concepts of ADT and its implementation for different types of data structures like Stack, Queue, and List.

CO2. Demonstrate the implementation of Stack, Queue and List for real world applications.

CO3. Demonstrate the usage and implementation of Tree and Binary Search Tree.

CO4. Describe and Demonstrate the concepts, algorithms and applications of AVL tree, Heaps and different operations on Multiway Trees and graphs.

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO1, PO2, PO3
CO 2	PO3
CO 3	PO3
CO 4	PO3

SEMESTER –II

ANALYSIS AND DESIGN OF ALGORITHMS

Subject Code	18MCA23	**CIE Marks**	30
Number of Lecture Hours/Week	4	**SEE Marks**	70
Total Number of Lecture Hours	50	**SEE Hours**	03

CREDITS – 4:0:0

Course Learning Objectives(CLO):

- Analyse the asymptotic performance of algorithms.
- To understand and apply algorithms and design techniques
- To solve problems using appropriate design techniques
- To analyze the complexities of various problems in different domains.
- Identify the limitations of algorithms power.

Modules	**Teaching Hours**
Module -1	**11 Hours**
Introduction, Fundamentals of the Analysis of Algorithm Efficiency Notion of Algorithm, Asymptotic Notations and Basic efficiency classes, Mathematical analysis of Recursive and Non-recursive algorithms. **Brute Force:** Selection Sort and Bubble Sort, Sequential Search and String Matching.	
Module -2	**10 Hours**
Divide-and-Conquer Merge sort, Quicksort, Binary Search, Multiplication of large integers **Greedy Technique** The General method, Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm, Huffman Trees.	
Module -3	**11 Hours**

Dynamic Programming Computing a binomial coefficient, Warshall's and Floyd's Algorithms, The Knapsack Problem **Decrease-and-Conquer** Depth First and Breadth First Search, Topological sorting. **Transform and Conquer** Introduction	
Module -4	**10 Hours**
Backtracking: n-Queens problem, Hamiltonian Circuit Problem, Subset – Sum Problem. **Branch-and-Bound**: Assignment Problem, Knapsack Problem	
Module -5	**10 Hours**
Limitations of Algorithm Power Lower-Bound Arguments, Decision Trees, P, NP and NP-Complete Problems. **Space and Time Tradeoffs** Sorting by Counting, Input Enhancement in String Matching.	

Question paper pattern:

- There will be 10 questions with 2 full questions from each module.
- Each full question consists of 20 marks.
- Students have to answer 5 full questions, selecting ONE from each module.

Text Books:

1. Anany Levitin: Introduction to the Design and Analysis of Algorithms, Pearson Education, 2nd Edition

Reference Books:

1. Coremen T.H., Leiserson C.E., and Rivest R.L.: Introduction to Algorithms, PHI 1998.
2. Horowitz E., Sahani S., Rajasekharan S.: Computer Algorithms, Galgotia Publication 2001.
3. Michael T Goodrich and Roberto Tamassia : Algorithm Design, Wiley India
4. R C T Lee, S S Tseng, R C Chang, Y T Tsai : Introduction to Design and Analysis of Algorithms: A Strategic Approach, Tata McGraw Hill

COURSE OUTCOMES(CO):

CO1: Analyse time and space complexity of recursive and non- recursive algorithms.

CO2: Analyze algorithms and solve real time problems using various algorithm design techniques.

CO3: Design and analyse algorithms to solve the optimization problems.

CO4: Design and analyze algorithms associated with space–time tradeoffs with the limitations of Algorithm power.

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO1,PO2
CO 2	PO1,PO2,PO3
CO 3	PO1,PO2,PO3
CO 4	PO2,PO3

SEMESTER –II

DATABASE MANAGEMENT SYSTEM

Subject Code	**18MCA24**	CIE Marks	**50**
Number of Lecture Hours/Week	**4**	SEE Marks	**50**
Total Number of Lecture Hours	**52**	SEE Hours	**03**

CREDITS – 4:0:0

Course Learning Objectives(CLO):

- Students understand and implement the processes of database management system.
- Students understand the design guidelines for a relational schema and learn to query from a database
- Students understand managing multiple transactions and recovery techniques in case of transaction failures

Modules	**Teaching Hours**
Module -1	**10 Hours**
Characteristics of Database approach, Actors on the Scene, Workers behind the scene, Advantages of using DBMS approach, Data models, schemas and instances, Three -schema architecture and data independence, Database languages and interfaces, the database system environment, Centralized and client -server architectures, Classification of Database Management systems, Entity-Relationship Model: Conceptual Database using high level conceptual data models for Database Design, A Sample Database Application, Entity types, Entity sets Attributes and Keys Relationship types, Relationship Sets, Roles and Structural Constraints Weak Entity Types..	
Module -2	
Relational Model: Relational Model Concepts, Relational Model Constraints and Relational Database Schema Update Operations, Transactions and Dealing with Constraint violations	**10 Hours**
Module -3	**10 Hours**
Introduction to SQL: Overview of the SQL Query Language, SQL Data Definition, Basic structure of SQL Queries, Additional Basic Operations, Null values, Aggregate Functions, nested Sub queries, Modification of the Database, Join Expressions, Views, Transactions, Integrity Constraints	

Module -4	**10 Hours**
Database Design: Informal Design Guidelines for Relation Schemas, Functional Dependencies, Normal Forms based on Primary Keys, General Definitions of 2nd and 3rd Normal Forms, Boyce Codd Normal Forms, Stored Procedures and functions, Triggers.	
Module -5	**12Hours**
Transaction Management: Transaction Concept, A Simple Transaction Model, Transaction Atomicity and Durability, Serializability, Transaction Isolation and Atomicity, Transaction Isolation Levels, Implementation of Isolation Levels. Concurrency Control: Lock Based Protocols, Deadlock Handling. Recovery System: Failure Classification, Storage, Recovery and Atomicity, Recovery Algorithm.	

Question paper pattern:

- There will be 10 questions with 2 full questions from each module.
- Each full question consists of 20 marks.
- Students have to answer 5 full questions, selecting ONE from each module.

Text Books:

- Elmasri and Navathe: Fundamentals of Database Systems, 5th Edition, Addison -Wesley, 2011.
- Silberschatz, Korth and Sudharshan Data base System Concepts,6th Edition, Tata McGraw Hill, 2011.

Reference Books:

1. C.J. Date, A. Kannan, S. Swamynatham: An Introduction to Database Systems, 8th Edition, Pearson education,2009.
2. 2 Raghu Ramakrishnan and Johannes Gehrke: Database management Systems, 3rd Edition, McGraw-Hill, 2003.

Course Outcome (CO):

CO1: Students are demonstrated on the fundamentals of data models and develop an ER diagram and relational database model for a given scenario

CO2: Students understand to query the database

CO3: Students apply the rules of normalization to Inference the database design in the real world entities, multiple transactions, concurrency control techniques and recovery.

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO1
CO 2	PO1, PO2,PO4
CO 3	PO1, PO2,PO4,PO5,PO7
CO 4	PO1, PO2,PO4,PO5,PO7, PO8

SEMESTER – II

Operating Systems

Subject Code	**18MCA24**	CIE Marks	**50**
Number of Lecture Hours/Week	**3**	SEE Marks	**50**
Total Number of Lecture Hours	**39**	SEE Hours	**03**

CREDITS – 3:0:0

Course Learning Objectives(CLO) :

- Understand the features and design of an Operating Systems
- Apply methods for handling of Process Management and Mutual Exclusion
- Analyze different approaches to memory management.
- Analyze file handling system

Modules	**Teaching Hours**
Module -1	**10 Hours**
Introduction to Operating Systems, System Structure What operating systems do; Computer System Organization; Computer System Architecture; Operating System Operations; Computing Environments; Operating System Services; System Calls; Types of System Calls; System Programs; Operating System Structure; Virtual Machines; **Overview of Process** **Process** Concept; Process Scheduling; Operations on Processes; Inter – Process Communication; Multi – Threaded Programming: Overview; Multithreading Models;	
Module -2	**10 Hours**
Process Management Process Scheduling: Basic Concepts, Scheduling Criteria, Scheduling Algorithms, Multiple Processor Scheduling **Process Synchronization** Synchronization: The Critical Section Problem: Peterson's Solution; Semaphores; Classical Problems of Synchronization;	
Module -3	**10 Hours**
Deadlocks Deadlocks: System model; Deadlock Characterization, Methods for handling deadlocks; Deadlock Prevention; Deadlock Avoidance; Deadlock Detection and Recovery from deadlock **Memory Management** Memory Management Strategies: Background, Swapping;	

Contiguous Memory Allocation; Paging; Segmentation; Virtual Memory Management; Background; Demand Paging; Page Replacement; Allocation of Frames; Thrashing	
Module -4	**9 Hours**
File System Implementation of File System: File concepts; Access methods, Directory Structure; File System Mounting; Protection; Implementing File System: File System Structure; File System implementation; Directory Implementation; Allocation Methods; Free Space Management	

Question paper pattern:

- There will be 10 questions with 2 full questions from each module.
- Each full question consists of 20 marks.
- Students have to answer 5 full questions, selecting ONE from each module.

Text Books:

1. . Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating Systems Principles, 8th Edition, Wiley – India.

Reference Books:

1. D M Dhamdhere: Operating Systems – A Concept Based Approach, 2nd Edition, Tata McGraw – Hill, 2002.
2. P C P Bhatt: Operating Systems, 2ndEdition, PHI, 2006.
3. Harvey M Deital: Operating Systems, 3rdEdition, Addison Wesley, 1990.

Course Outcome (CO):

CO1: Describe the elements and various functionalities of the operating system to a basic level
CO2: Illustate various memory allocation strategies and implement virtual memory techniques for effective memory management
CO3: Apply methods for process scheduling, process synchronization, and deadlock handling
CO4: Analyze the physical and logical structure of the storage media, illustrate various algorithms for storage management

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO1
CO 2	PO1
CO 3	PO1
CO 4	PO1

SEMESTER –II

JAVA PROGRAMMING LABORATORY

Laboratory Code	**18MCAL26**	CIE Marks	**50**
Number of Lecture Hours/Week	**02**	SEE Marks	**50**
		SEE Hours	**03**

CREDITS – 0:0:1

Course Learning Objectives(CLO) :

- Design & Develop the fundamentals of Object-oriented programming in Java, including defining classes, invoking methods, using class libraries.
- Design & Develop exception handling and multithreading concepts.
- Develop efficient Java applets and applications using OOP concepts.
- Design & Develop basic understanding of network application programs.

1. A)Write a JAVA Program to demonstrate Constructor Overloading and Method Overloading.

   b) Write a JAVA Program to implement Inner class and demonstrate its Access protection.

2. Write a program in Java for String handling which performs the following:

   i) Checks the capacity of String Buffer objects.

   ii) Reverses the contents of a string given on console and converts the resultant string in upper case.

   iii) Reads a string from console and appends it to the resultant string of (ii).

3. a). Write a JAVA Program to demonstrate multi-level Inheritance.

   b). Simple Program on Java for the implementation of Multiple inheritance using interfaces to calculate the area of a rectangle and triangle.

4. Write a JAVA program which has

   i) A Class called Account that creates account with 500Rs minimum balance, a deposit()method to deposit amount, a withdraw() method to withdraw amount and also throws Less Balance Exception if an account

holder tries to withdraw money which makes the balance become less than 500Rs.

ii) A Class called LessBalanceException which returns the statement that says withdraw amount (Rs) is not valid.

iii) A Class which creates 2 accounts, both account deposit money and one account tries to withdraw more money which generates a LessBalanceException take appropriate action for the same.

5. Write a java program to handle the following system exceptions
   ArrayIndexOutOfBoundException
   FileNotFoundException
   NumberFormatException

6. a) Write a JAVA program using Synchronized Threads, which demonstrates Producer Consumer concept.
   b) Design a program to create two threads, one thread will print odd numbers and second thread will print even numbers between 1 to 10 numbers

7. Write a JAVA program to implement a Queue using user defined Exception Handling (also make use of throw, throws).

8. Complete the following:
   i. Create a package named shape.
   ii. Create some classes in the package representing some common shapes like Square, Triangle, and Circle.
   iii. Import and compile these classes in other program.

9. Write a JAVA program which has
   i). A Interface class for Stack Operations
   ii). A Class that implements the Stack Interface and creates a fixed length Stack.
   iii). A Class that implements the Stack Interface and creates a Dynamic length Stack.
   iv). A Class that uses both the above Stacks through Interface reference and does the Stack operations that demonstrates the runtime binding.

10. Write a JAVA Program which uses FileInputStream / FileOutPutStream Classes.

11. Write JAVA programs which demonstrates utilities of LinkedList Class and Array class.

12. Write a JAVA program which uses Datagram Socket for Client Server Communication for multiple systems

PART B

(Mini Project)

Student should carryout menu driven interactive program on any simple real world problem using JAVA. Students may do the above project with GUI as well.

Note 1: In the practical Examination student has to execute one program from a lot of all the 12 questions and demonstrate Part B Mini Project.

Note 2: Project to be carried out with a team of 2 or individual student

Note 3: Change of program is not permitted in the Practical Examination

Course Outcomes(CO):

CO1: Understand Java programming language fundamentals and runtime environment. Gain knowledge and skill necessary to write java programs. Learn the object oriented concepts and its implementation in Java implement the multithreading and client side programming

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO1,PO2,PO3,PO4,PO5,PO7,PO8,PO11

SEMESTER –II

DATA STRUCTURES LABORATORY

Laboratory Code	**18MCAL27**	CIE Marks	**50**
Number of Lecture Hours/Week	**02**	SEE Marks	**50**
		SEE Hours	**03**

CREDITS – 0:0:1

Course Learning Objectives(CLO) :

- Design the programming codes for the implementation of data structure concepts.
- Develop the programming codes to illustrate the applications of data structures.
- Develop the programming codes to illustrate the advanced concepts of Tree

1. Design, Develop and Implement a menu driven program in C++ for the following operations on STACK of integers (Array implementation of stack with maximum size MAX)

i. Push an element on to stack
ii. Pop an element from stack.
iii. Demonstrate how stack can be used to solve tower of Hanoi problem with n disks.
iv. Demonstrate Overflow and Underflow situations on stack.
v. Display the status of stack.
vi. Exit.

Support the program with appropriate functions for each of the above operations.

2. Design, develop and Implement a Program in C++ for converting an Infix Expression to Postfix Expression. Program should support for both parenthesized and free parenthesized expressions with the operators: +, -, *, /, %(Remainder), ^ (Power) and alphanumeric operands.

3. Design, Develop and Implement a Program in C++ for the following Stack Applications

i. Evaluation of Suffix expression with single digit operands and operators: +, -, *, /, %, ^
ii. Generate Fibonacci Numbers.

4. Design, Develop and Implement a menu driven Program in C++ for the following operations on Circular QUEUE of Characters (Array Implementation of Queue with maximum size MAX).

i. Insert an Element on to Circular QUEUE
ii. Delete an Element from Circular QUEUE
iii. Demonstrate Overflow and Underflow situations on Circular QUEUE

iv. Display the status of Circular QUEUE
v. Exit

Support the program with appropriate functions for each of the above operations

5. Design, Develop and Implement a menu driven Program in C++ for the following operations on Singly Linked List (SLL) of Student Data with the fields: USN, Name, Branch, Semester, Mobile number

i. Create a SLL of N Students Data by using front insertion.
ii. Display the status of SLL and count the number of nodes in it
iii. Perform Insertion and Deletion at End of SLL
iv. Perform Insertion and Deletion at Front of SLL
v. Demonstrate how this SLL can be used as STACK and QUEUE
vi. Exit

6. Design, Develop and Implement a menu driven Program in C++ for the following operations on Doubly Linked List (DLL) of Employee Data with the fields: SSN, Name, Dept, Designation, Sal, PhNo.

i. Create a DLL of N Employees Data by using end insertion.
ii. Display the status of DLL and count the number of nodes in it
iii. Perform Insertion and Deletion at End of DLL
iv. Perform Insertion and Deletion at Front of DLL
v. Demonstrate how this DLL can be used as Double Ended Queue
vi. Exit

7. Design, Develop and Implement a Program in C++ for create insert, delete and exit operations on Singly Circular Linked List (SCLL) with header nodes

8. Design, Develop and Implement a menu driven Program in C++ for the following operations on Binary Search Tree (BST) of Integers

i. Create a BST of N Integers
ii. Traverse the BST in Inorder, Preorder and Post Order
iii. Search the BST for a given element (KEY) and report the appropriate message
iv. Delete an element (ELEM) from BST
v. Exit

9. Design, develop and Implement a menu driven Program in C++ to perform create, insert, delete and display operations on AVL Tree.

10. Design, develop and Implement a menu driven Program in C++ to perform create, insert, delete and display operations on Heap Tree.

11. Design, develop and Implement a menu driven Program in C++ to perform create, insert, delete and display operations on M-way search trees.

12. Design, develop and Implement a menu driven Program in C++ to perform create, insert, delete and display operations on B-Trees.

Course Outcome(CO):

CO1: **Design, Develop and Implement various applications of data structures.**

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO1,PO2,PO3,PO4,PO5,PO7,PO8,PO11

SEMESTER –II

DATABASE MANAGEMENT SYSTEMS LAB

Laboratory Code	**18MCAL28**	CIE Marks	**50**
Number of Lecture Hours/Week	**02**	SEE Marks	**50**
		SEE Hours	**03**

CREDITS – 0:0:1

Course Learning Objectives(CLO):

- Demonstrate SQL commands.
- Implement simple exercises on relational database schema.
- Design a relational database schema for specific database application using SQL.

DBMS Lab Experiments:

Instructions for the Exercises:

1. Draw an ER diagram based on given scenario.

2. Create Relational Database Schema based on the above scenario using Mapping Rules.

3. Perform the given queries using any RDBMS Environment.

4. Suitable tuples have to be entered so that queries are executed correctly

5. The results of the queries may be displayed directly

1. **Exercise: 1**

Notown Records has decided to store information about musicians who perform on its albums. Each musician that records at Notown has an SSN, a name, an address, and a phone number. No musician has more than one phone. Each instrument used in songs recorded at Notown has a unique identification number, a name (e.g., guitar, synthesizer, flute) and a musical key (e.g., C, B-flat, E-flat). Each album recorded on the Notown label has a unique identification number, a title, a copyright date and a format (e.g., CD or MC). Each song recorded at Notown has a title and an author. Each musician may play several instruments, and a given instrument may be played by several musicians. Each album has a number of songs on it, but no song may appear on more than one album. Each song is performed by one or more musicians,

and a musician may perform a number of songs. Each album has exactly one musician who acts as its producer. A musician may produce several albums.

Queries

a) List musician name, title of the song which he has played, the album in which song has occurted.

b) List the details of songs which are performed by more than 3 musicians.

c) List the different instruments played by the musicians and the average number of instruments played.

d) Retrieve album title produced by the producer who plays guitar as well as flute e) List the details of musicians who can play all the instruments present.

Exercise2

Professors have a PROFID, a name, an age, a rank, and a research specialty. Projects have a project number, a sponsor name (e.g. UGC/AICTE/…), a starting date, an ending date, and a budget. Graduate students have an USN, a name, an age, and a degree program (e.g. MCA/MPhil/BE/ME ..). Each project is managed exactly by one professor (known as the project's principal investigator). Each project is worked on by one or more professors (known as the project's co-investigators). Professors can manage/work on multiple projects. Each project is worked on by one or more graduate students (known as the project's research assistants). Graduate students can work on multiple projects. Each professor can supervise many students. A student who is working on a project can be supervised by only one professor.

Queries

a) Retrieve the names of all professors who do not have an ongoing project of more than 1 lakh.

b) Retrieve the names of all graduate students along with their professors under whom they work and project sponsor.

c) List the professors and sum of the budget of their projects started after 2005 but ended in 2010.

d) List the names of professors who has a total worth of project greater than the average budget of projects sanctioned

e) List the professors who work on all the projects.

Exercise: 3

A bank has many branches and a large number of customers. Bank is identified by its code. Other details like name, address and phone for each bank are also stored. Each branch is identified by its bank. Branch has name, address and phone. A customer can open different

kinds of accounts with the branches. An account can belong to more than one customer. Customers are identified by their SSN, name, address and phone number. Age is used as a factor to check whether customer is a major. There are different types of loans, each identified by a loan number. A customer can take more than one type of loan and a loan can be given to more than one customer. Loans have a duration and interest rate. Make suitable assumptions and use them in showing maximum and minimum cardinality ratios.

Queries:

a) List the details of customers who have joint account and also have at least one loan.

b) List the details of the branch which has given maximum loan.

c) List the details of saving accounts opened in the SBI branches located at Bangalore.

d) List the name of branch along with its bank name and total amount of loan given by it.

e) Retrieve the names of customers who have accounts in all the branches located in a specific city.

Exercise: 3

A bank has many branches and a large number of customers. Bank is identified by its code. Other details like name, address and phone for each bank are also stored. Each branch is identified by its bank. Branch has name, address and phone. A customer can open different

kinds of accounts with the branches. An account can belong to more than one customer. Customers are identified by their SSN, name, address and phone number. Age is used as a factor to check whether customer is a major. There are different types of loans, each identified by a loan number. A customer can take more than one type of loan and a loan can be given to more than one customer. Loans have a duration and interest rate. Make suitable assumptions and use them in showing maximum and minimum cardinality ratios.

Queries:

a) List the details of customers who have joint account and also have at least one loan.

b) List the details of the branch which has given maximum loan.

c) List the details of saving accounts opened in the SBI branches located at Bangalore.

d) List the name of branch along with its bank name and total amount of loan given by it.

e) Retrieve the names of customers who have accounts in all the branches located in a specific city.'

Exercise: 4

Patients are identified by an SSN, and their names, addresses, and ages must be recorded. Doctors are identified by an SSN. For each doctor, the name, specialty, and years of experience must be recorded. Each pharmaceutical company is identified by name; it has an address and one phone number. For each drug, the trade name and formula must be recorded. Each drug is sold by a given pharmaceutical company, and the trade name identifies a drug uniquely from among the products of that company. Each pharmacy has a name, address, and phone number. Each patient is checked up by some doctor. Every doctor has at least one patient. Each pharmacy sells several drugs and has a price for each. A drug could be sold at several pharmacies, and the price could vary from one pharmacy to another. Doctors prescribe drugs for patients. A doctor could prescribe one or more drugs for several patients, and a patient could obtain prescriptions from several doctors. Each prescription has a date and a quantity associated with it. Pharmaceutical companies have long-term contracts with pharmacies. A pharmaceutical company can contract with several pharmacies, and a pharmacy can contract with several pharmaceutical companies. For each contract, you have to store a start date, an end date, supervisor and the text of the contract.

Queries:

a) List the details of patients who are 20 years old and have been checked by eye-specialist.

b) List the details of doctors who have given the prescription to more than 20 patients in year 2013.

c) List the details of pharmaceutical companies who supply drug to more than 10 pharmacies d) List the details of drug supplied by only one pharmaceutical company.

e) List the details of drug supplied by all pharmaceutical companies.

Exercise; 5

Data requirements of movie industry are captured. Each movie is identified by title and year of release. Each movie has length in minutes and classified under one genres (like action, horror etc.). Each movie has a plot outline. Production companies are identified by name and each has an address. A production company produces one or more movies. Actors are identified by id. Other details like name and date of birth of actors are also stored. Each actor acts in one or more movies. Each actor has a role in movie. Directors are identified by id. Other details like name and date of birth of directors are also stored. Each director directs one or more movies. Each movie has one or more actors and one or more directors and is produced by a production company.

Queries:

a) List the details of horror movies released in 2012 and directed by more than 2 directors.

b) List the details of actors and total number of movies acted between 2000 and 2010.

c) List the details of production companies producing maximum movies.

d) List the details of movies where director and actor have same date of birth.

e) Retrieve the names of directors directed all the movies produced by any one production company.

Course Outcome (CO):

CO1: Design an ER diagram and implement a database schema for a given problem domain and query the tables.

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO1,PO2,PO3,PO4, PO7

Dr. Ambedkar Institute of Technology, Bengluru-56

(An Autonomous Institute, Approved by AICTE, Affiliated to V T U, Belagavi)
Nationally Accredited by NAAC with 'A' Grade

Master of Computer Applications

(Accredited by National Board of Accreditation)

MCA III to IV Semester

(2018-21) Scheme & Syllabus

		III Semester							
Subject Code	**Name of the Subject**	**Credits**				**Exam Duration in Hours**	**Marks**		**Total Marks**
		Lecture	**Tutorial**	**Practicals**	**Total Credits**		**CIE**	**SEE**	
18MCA31	Network Architecture & Management	4		-	4	3	50	50	100
18MCA32	Python Programming	4		-	4	3	50	50	100
18MCA33	Advanced Software Engineering	4		-	4	3	50	50	100
18MCA34	Data Science using R	3		-	3	3	50	50	100
18MCA35	Elective-1	3		-	3	3	50	50	100
18MCAL36	Network Architecture Lab	--	-	1	1	3	50	50	100
18MCAL37	Python programming Lab	--	-	1	1	3	50	50	100
18MCAL38	Data Science using R Lab	--	-	1	1	3	50	50	100
Total		**18**	**-**	**3**	**21**	**24**	**400**	**400**	**800**

ELECTIVE-1									
Subject Code	**Name of the Subject**	**Credits**				**Exam Duration**	**Marks**		**Total Marks**
		L	**T**	**P**	**Credits**		**CIE**	**SEE**	
18MCA351	Software Testing and Practices	3	-	-	3	3	50	50	100
18MCA352	Advanced DBMS	3	-	-	3	3	50	50	100
18MCA353	Artificial Intelligence	3	-	-	3	3	50	50	100

		IV Semester							
Subject Code	Name of the Subject	Credits				Exam Duration in Hours	Marks		Total Marks
		Lecture	Tutorial	practicals	Total Credits		CIE	SEE	
18MCA41	Enterprise Applications-1	3		-	3	3	50	50	100
18MCA42	Advanced Web Technologies	3		-	3	3	50	50	100
18MCA43	Analytical Skills And Building Professional Development	3		-	3	3	50	50	100
18MCA44	Elective-2	3		-	3	3	50	50	100
18MCA45	Elective-3	3		-	3	3	50	50	100
18MCA46	Research Methodology	2	-	--	2	3	50	50	100
18MCAL47	Enterprise Applications Lab	--	-	1	1	3	50	50	100
18MCAL48	Advanced Web Technologies Lab	--	-	1	1	3	50	50	100
18MCAM49	Mini Project using Android	--	1	1	2	3	50	50	100
Total		**17**	**1**	**3**	**21**	**27**	**450**	**450**	**900**

ELECTIVE-2									
Subject Code	**Name of the Subject**	**Credits**				**Exam Duration**	**Marks**		**Total Marks**
		L	**T**	**P**	**Credits**		**CIE**	**SEE**	
18MCA441	Information Security	3	-	--	3	3	50	50	100
18MCA442	Data Mining & Business Intelligence	3	-	--	3	3	50	50	100
18MCA443	Object oriented Modelling & Design	3	-	--	3	3	50	50	100

ELECTIVE-3									
Subject Code	**Name of the Subject**	**Credits**				**Exam Duration**	**Marks**		**Total Marks**
		L	**T**	**P**	**Credits**		**CIE**	**SEE**	
18MCA451	Software Quality & Performance Evaluation	3	-	--	3	3	50	50	100
18MCA452	Software Architecture	3	-	--	3	3	50	50	100
18MCA453	Enterprise Resource Planning	3	-	--	3	3	50	50	100

SEMESTER – III

Network Architecture and Management

Subject Code	**18MCA31**	CIE Marks	**50**
Number of Lecture Hours/Week	**4**	SEE Marks	**50**
Total Number of Lecture Hours	**52**	SEE Hours	**03**

CREDITS – 4:0:0

Course Learning Objectives (CLO):

- Classify the network protocols and identify the services
- Demonstrate the architecture of advanced networks and larger networks
- Analyze the significance of network management and apply the security tactics

Modules	**Teaching Hours**
Module -1	**12 Hours**
Protocols And Standards TCP/IP Reference Model, Circuit Switching, Packet Switching, Internet Protocols (TCP, IP, UDP, ICMP, ARP), Unicast Routing Protocols (OSPF, DVRP). Multicast Routing protocols (MOSPF, DVMRP)	
Module -2	**8 Hours**
Larger Networks Optical Networking: SONET/SDH, ATM, Frame Relay, Virtual LAN	
Module -3	**9 Hours**
Wireless Networking SAN, Bluetooth, wireless networks, Wireless sensor networks	
Module -4	**12 Hours**
Emerging Network Technologies Software Defined Network, Delay-tolerant Networking, Home Networking, Content Distributed Network, Data Center Network	
Module -5	**12 Hours**
Network Security Quality of Service – Techniques, Security: Firewalls: Packet Filtering, Stateful, Application Proxy, Spoofing: TCP, DNS, IP and Web Denial of Service: Single Source and distributed, Designing a comprehensives security plan	

Question paper pattern:

- There will be 2 full questions from each module.
- Each full question consists of 20 marks.
- Students have to answer 5 full questions selecting ONE from each module

Text Books:

1. A Practical Guide To Advanced Networking By Jeffrey S. Beasley And Piyasat Nilkaew, Pearson Education 2013 Ed
2. Jan L Harrington , Network Security: A Practical Approach, Morgan Kauffman, 2005

Reference Books:

- "Douglas E Comer, "Internetworking with TCP/IP, Principles, Protocols and Architecture" 6th Edition, PHI – 2014, ISBN-10: 0130183806
- Uyless Black "Computer Networks, Protocols, Standards and Interfaces" 2nd Edition - PHI , ISBN-10: 8120310411
- M. E. Whitman and Herbert J. Mattored, Principles of Information Security, Information Security Professional 4thedition.
- Tanenbaum, A., Computer Networks, 3rd ed., Prentice-Hall, 1996. Wright and Stevens, TCP/IP Illustrated v.2, Addison-Wesley

Course Outcomes(CO):

CO 1: Understand the network protocols and its services.

CO 2: Analyze the architecture of wired and wireless network variants

CO 3: Demonstrate the working architecture of emerging networks

CO 4: Analyze and Apply methodologies to build a secured network

COs	**Mapping with POs**
CO1	PO1, PO5
CO2	PO1, PO3, PO5, PO7
CO3	PO1, PO3, PO5
CO4	PO3, PO5, PO7

SEMESTER –III Python Programming			
Subject Code	**18MCA32**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**4**	**SEE Marks**	**50**
Total Number of Lecture Hours	**52**	**SEE Hours**	**03**
CREDITS – 4:0:0			

Course Learning Objectives (CLO):

- To describe the Fundamentals of Python
- To demonstrate the python data structure
- To implement files and data base connectivity and object oriented programing
- To develop Web applications using python

Modules	**Teaching Hours**
Module -1	**10Hours**
Overview of Python Introduction to Python: Features of Python, Execution of a Python Program, Viewing the Byte Code, Flavors of Python, Python Virtual Machine (PVM, Frozen Binaries, Memory Management in Python, Garbage Collection in Python, Comparisons between C and Python. Datatypes in Python, operators and I/O Statements: Comments in Python, User-defined Datatypes, Output statements, Input Statements Control Statements: Functions: Pass by Object Reference, Formal and Actual Arguments, Positional Arguments, Keyword Arguments, Default Arguments, Variable Length Arguments, Local and Global Variables, The Global Keyword, Passing a Group of Elements to a Function, Recursive Functions, Anonymous Functions or Lambdas,	
Module -2	**10 Hours**
Exception Handling and Regular expressions Exceptions: Errors in a Python Program, Exceptions, Exception Handling, Types of Exceptions, -The Except Block, The assert Statement, User-Defined Exceptions. Regular Expressions: Sequence Characters in Regular Expressions, Quantifiers in Regular Expressions, Special Characters in Regular Expressions, Using Regular Expressions on Files, Retrieving Information from a HTML File.	
Module -3	**12 Hours**

Python Data Structure Strings and Characters: Creating Strings, Length of a String, Indexing in 10 Strings, Slicing the Strings, Repeating the Strings, Concatenation of Strings, Hours String Methods, Lists, Tuples and Sets: , Nested Lists as Matrices, List Comprehensions, Tuples, Dictionaries: Operations on Dictionaries, Dictionary Methods, , Sorting the Elements of a Dictionary using Lambdas, Converting Lists into Dictionary, Converting Strings into Dictionary, Passing Dictionaries to Functions, Ordered Dictionaries. Using zip() Function, Sets, Set Methods, Traversing of Sets,	
Module -4	**10 Hours**
Files and database Connectivity Files: Types of Files in Python, Opening a File, Closing a File, Working with 10 Text Files Containing Strings, Knowing Whether a File Exists or Not, Hours Working with Binary Files, The with Statement, Pickle in Python, The seek() and tell() Methods, Python's Database Connectivity: Types of Databases Used with Python, Working with MySQL Database, Using MySQL from Python, Retrieving All Rows from a Table, Inserting Rows into a Table, Deleting Rows from a Table, Updating Rows in a Table, Creating Database Tables through Python	
Module -5	**10 Hours**
Object Oriented Programming Concepts OOP in Python: Specialty of Python Language, Creating a Class, The Self 10 Variable, Constructor, Types of Variables, Namespaces, Types of Methods, Hours Passing Members of One Class to Another Class, Inner Classes. Inheritance and Polymorphism, Constructors in Inheritance, Overriding Super Class, Constructors Inheritance, Overriding Super Class, Constructors and Methods, The super() Method, Types of Inheritance, Method Resolution Order (MRO), Polymorphism, Operator Overloading, Method Overloading, Method Overriding, Abstract Classes and Interfaces, Abstract Method and Abstract Class, Interfaces in Python, Abstract Classes vs. Interfaces. **Introduction to Networking concepts in python** Python Network services, socket program, simple networking programs.	
Question paper pattern: • The question paper will have five questions. All questions are compulsory. Module 1 and 5 shall have internal choice. • Each full question consists of 20 marks. Questions are set covering all the topics under a each module.	

Text Books:

1. Core Python Programming: 2017 Edition, R. Nageswara Rao, DreamTech Publication.
2. Exploring Python, Timothy A. Budd, Mc Graw Hill Education
3. Introduction to Python Programming , Gowrihankar S, Veena A, CRC Press/Tyler and Francies.
4. Practical Programming: An introduction to Computer Science Using Python, second edition, Paul Gries, Jennifer Campbell, Jason Montojo, The Pragmatic Bookshelf.

Reference Books:

1. Introduction to Python for Computational Science and Engineering (A beginner's guide), Hans Fangohr.
2. Python for Informatics: Exploring Information, Charles Severance.
3. Learning Python, Fourth Edition, Mark Lutz, O'Reilly publication.

Course Outcome (CO):

CO1: Understand the Fundamentals of Python programming

CO2: Demonstrate various features of python programming for building Applications.

CO3: Apply python programming for designing the applications efficiently.

CO4: Design and Develop applications to be deployed in real world scenarios.

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO1,PO2,PO3,PO4,PO5,PO8
CO 2	PO1,PO2,PO3,PO4
CO 3	PO1,PO2,PO3,PO4,PO5
CO 4	PO1,PO2,PO3,PO4,PO5

SEMESTER –III Advanced Software Engineering			
Subject Code	**18MCA33**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**4**	**SEE Marks**	**50**
Total Number of Lecture Hours	**52**	**SEE Hours**	**03**
CREDITS – 4:0:0			

Course Learning Objectives (CLO):

- Understand the software life cycle models and International standards of software engineering
- Analyze the software requirements engineering
- Demonstrate the limitations of Software Design
- Design and develop correct and robust software products
- Identify the Software Testing, Software Quality and Software Maintenance

Modules	Teaching Hours
Module -1	**12Hours**
Introduction: Professional Software Development Attributes of good software, software engineering diversity, IEEE/ ACM code of software engineering ethics, case studies **Software Engineering and its models:** Evolution of Software Engineering Software development models, Capability maturity models, Software process technology **Software Process & Agile Software Development** Software Process models: waterfall, incremental development Agile methods, Plan-driven and agile Development, Extreme Programming, Agile project management, Scaling agile methods.	
Module -2	**12 Hours**
Requirements Engineering and System models Requirements engineering: Functional and Non-functional requirements, the software requirements document, Requirements specification, Requirements engineering processes, Requirements elicitation and analysis, Requirements validation, Requirements management. System modeling: Context models, Interaction Models, Structural Models, Behavioral models, Model-driven engineering.	
Module-3	**10 Hours**
Software design concepts and principles Design characteristics, Architectural Design, Architectural views, Architectural design patterns, System structuring, Control models;	

Structured design; Object-oriented analysis and design; User interface design; Design for reuse	
Module -4	**10 Hours**
Software Implementation Structured coding Techniques, Coding Styles, Standards and Guidelines, Documentation Guidelines, Modern Programming Language Features: Type checking, User defined data types, Data Abstraction, Exception Handling, Concurrency Mechanism.	
Module -5	**10 Hours**
Software Testing, Project Planning and Management Software Testing: Development testing, Test driven development, Release testing, User testing; Project Planning and Management: Risk management, Plan-driven development, Project scheduling, Estimation techniques; cost and effort estimation for the problem identified.	

Question paper pattern:

- The question paper will have five questions. All questions are compulsory. Module 1 and 5 shall have internal choice.
- Each full question consists of 20 marks. Questions are set covering all the topics under each module.

Text Books:

1. Ian Sommerville : Software Engineering, 9th edition, Person Education Ltd, 2011. (Chapters-: 1, 2, 3, 4, 5, 17, 18)
2. PankajJalote: Software Engineering, Wiley India Pvt Ltd (2010) (Chapters-:4, 6.1, 6.2, 6.5, 6.6)

Reference Books:

1. Roger S Pressman: Software Engineering-A Practitioners approach, 6th Edition, McGraw-Hill, 2010.
2. Hans Van Vliet: Software Engineering Principles and Practices, 3rd Edition, Wiley–India, 2010

Course Outcome (CO):

CO1: Understand the Software Development Life cycle and Professional ethics.

CO2: Demonstrate the Requirements Engineering Process

CO3: Design and develop Software Models to develop robust software products

CO4: Illustrate the Software Implementation Standards and Techniques

CO5: Evaluate Software testing, Project planning and Project Management.

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO6
CO 2	PO9,PO10,PO11
CO 3	PO3, PO11,PO12
CO 4	PO8, PO11,PO12
CO 5	PO9, PO10,PO11

SEMESTER – III

Data Science using R

Subject Code	**18MCA34**	CIE Marks	**50**
Number of Lecture Hours/Week	**3**	SEE Marks	**50**
Total Number of Lecture Hours	**39**	SEE Hours	**03**

CREDITS – 3:0:0

Course Learning Objectives (CLO):

- To understand the concepts of Data science
- To analyse the sampling techniques for data classification.
- To implement modeling methods for machine learning problems
- Analyzing data from files and Visualizing graphical presentations using R

Modules	**Teaching Hours**
Module -1	**8 Hours**
DATA SCIENCE PROCESS Introduction, Evolution of data science, Data science process – roles, stages in data science project – components of the Data Science lifecycle, data analytics, exploring data – managing data – cleaning and sampling for modeling and validation	
Module -2	**8 Hours**
EXPLORING R BASICS Introduction, R features basic data types, Vectors, Lists ,Arithmetic, logical & Matrix Operations, Control structures, Functions in R, Data frames ,Reading Data & cleaning data Data visualization techniques –Histograms, box plot, bar chart, scatter plot.	
Module -3	**7 Hours**
STATISTICAL MEASURES IN R Understanding Descriptive statistics, standard deviations, probability distribution, Normal distribution-Skewness, kurtosis, Inferential statistics-Hypothesis testing- t-test -One Sampled t-test, Correlation - Person correlation coefficient.	
Module -4	**8 Hours**
DATA SCIENCE ALGORITHMS Classification of Regression Analysis, Regression process, Linear Regression, Understanding Linear regression, making prediction-hypothesis on regression coefficients, multiple Linear Regression, concepts and formulas, Logistic regression, Model building and making predictions, Adding best fit.	
Module -5	**8 Hours**

MODELLING TECHNIQUES Classification techniques-Decision Trees, K Nearest Neighbor classification Technique. Implementation in R Clustering techniques, Applications, k-means Clustering algorithm, Performance of k-means, choosing Initial centroid- Implementation in R, Efficiency using Confusion matrix	

Question paper pattern:

- There will be 2 full questions from each module.
- Each full question consists of 20 marks.
- Students have to answer 5 full questions selecting ONE from each module.

Text Books:

1. Nina Zumel, John Mount, "Practical Data Science with R", Manning Publications, 2014.
2. David Dietrich , Barry Heller ,"Data Science & Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data",Wiley,2015
3. Joseph Schmuller, "Statistical Analysis with R", John Wiley, 2017.

Reference Books:

1. W. N. Venables, D. M. Smith and the R Core Team, "An Introduction to R", 2013.
2. Pang-Ning Tan, Michael Steinbach, Vipin Kumar: Introduction to Data Mining, Addison- Wesley, 2005

Course Outcomes(CO):

CO 1: Understand role and process of Data Science.

CO 2: Apply exploratory methods for statistical modeling and analysis using R.

CO 3: Analyse modelling methods and interpret the results visually.

CO 4: Construct use cases to validate approach and identify modifications required.

Course Outcomes(CO)	**Mapping with Pos**
CO1	PO1,PO2
CO2	PO1,PO2,PO4,PO5
CO3	PO1,PO2,PO4,PO5,PO7
CO4	PO4,PO5,PO7

SEMESTER – III

Software Testing And Practices

Subject Code	**18MCA351**	CIE Marks	**50**
Number of Lecture Hours/Week	**3**	SEE Marks	**50**
Total Number of Lecture Hours	**39**	SEE Hours	**03**

CREDITS – 3:0:0

Course Learning Objectives (CLO):

- The process of Software Testing Life Cycle and Types of Testing
- Differentiate between Manual Testing and Automation in Testing.
- Design Test Cases using the testing tool Selenium IDE and Web Driver
- Identify different web Elements and apply them to design test cases

Modules	Teaching Hours
Module -1	**7 Hours**
Introduction To Testing Introduction and Fundamentals of Testing, Myths and Facts of Software Testing, Quality Assurance and Quality Control, Testing Objectives, Software Testing Life Cycle (STLC), Test Planning, Test-case Design Technique. Types of Testing- White Box testing, Black Box Testing, Integration Testing, Regression Testing, Validation Testing, Alpha Testing, Beta Testing, Acceptance Testing. Defect Management: Defect Management Process, JIRA Defect Tracking Tools.	
Module -2	**8 Hours**
Overview Of Selenium Software Test Automation: Fundamentals of Test Automation, Manual Testing Vs Test Automation, Terms used in Automation, Skills needed for Automation, Scope of Automation, Challenges in Automation. Selenium –WebDriver: Introduction to WebDriver , Installation of Selenium WebDriver, Creating the Scripts in WebDriver, Web Element Locators, Xpath , id, LinkText, CSS Selector, Class Name, TagName WebDriver Commands: Browser Commands, getUrl(), getTitle(), getPageSource(),close(),Quit(), Navigation Commands, backward(0,forward(),to(),refresh() WebElements Commands, Isselected(), IsEnabled(), getAttribute(), getText(),notify(), sendKeys(), submit(), wait().	
Module -3	**8 Hours**
Handling Webelements CheckBox and Radio Button Operation, DropDown and Multiple select Operations, Handle Alert in WebDriver : dismiss(), accept(), getText(), sendKey(), Popup window handling in Web Drivers, getWindowHandle(), Wait commands in Web Drivers, Mouse Event using Action commands, Handling Multiple Windows, and IFrames, Running Test in Invisible Mode, Handling Dynamic Web Pages. Running selenium Web Driver in different popular Browser	

Module -4	8 Hours
Application Programming Interface(API) Testing Introduction, API testing types, Command, tests on APIs, Advantages, Tasks involved, Tools used for API Testing, Challenges, Best Practices, Case study, Database Testing, Security Testing	
Module -5	**8 Hours**
Advanced Topics on Testing Cross Platform Testing, coding standard overview, code coverage metrics, code freeze, code Inspection, code Review, code walkthrough, code based testing, code driven Testing, CUCUMBER Framework, Test Driven development, Behavioral driven Development.	

Question paper pattern:

- There will be 2 full questions from each module.
- Each full question consists of 20 marks.
- Students have to answer 5 full questions selecting ONE from each module.

Text Books:

1. Rex Black: Advanced Software Testing—Vol. 1, Shroff Publishers, 2011.
2. Srinivasan Desikan Gopalaswamy: Software Testing Principles and Practices, 5th Edition, Pearson Education, 2007.
3. Paul C. Jorgensen: Software Testing A Craftman's Approach, 3rd Edition, Auernac Publications, 2008
4. David Burns: Selenium 2 Testing Tools: Beginner's Guide, Packt Publishing, 2012.

Reference Books:

1. Rex Black: Advanced Software Testing—Vol. 2, Shroff Publishers, 2011.
2. Gundecha Unmesh: Selenium Testing Tools Cook Book, PACKT PUBLISHING, 2012.

Course Outcomes(CO):

CO 1: Analyze the process of Software Testing Life Cycle and types of Testing.
CO 2: Demonstrate Manual Testing and Automation in Testing

CO 3: Design Test Cases for User Interface Testing
CO 4: Design Test Cases for Application Programming Interface (API) Testing and Data base Testing

Cos	Mapping with Pos
CO1	PO1, PO5, PO10
CO2	PO1, PO2, PO3, PO4
CO3	PO1, PO2, PO4, PO5, PO8
CO4	PO1, PO2, PO3, PO11, P12

SEMESTER – III Advanced DBMS			
Subject Code	**18MCA352**	CIE Marks	**50**
Number of Lecture Hours/Week	**3**	SEE Marks	**50**
Total Number of Lecture Hours	**39**	SEE Hours	**03**
CREDITS – 3:0:0			

Course Learning Objectives (CLO):

- Acquire the conceptual knowledge on various databases
- Learn NoSQL Database
- Apply database concepts in distributed applications
- Understand the role of distributed databases in modern applications

Modules	Teaching Hours
Module -1	**7 Hours**
Overview Comparison between different databases: Significance of Databases, Database System Applications, Advantages and Disadvantages of different Database Management systems, Comparison between DBMS, RDBMS, Distributed and Centralized DB.	
Module -2	**8 Hours**
Working With Nosql Introduction to NoSQL, Types of NoSQL Databases, Advantages of NoSQL, SQL versus NoSQL, Aggregate Data Models, Aggregates, Key-value and Document Data Models, Relationships, Graph Databases, Schema-less Databases, Distribution Models, CAP Theorem, Eventual Consistency, Advantages and Disadvantages of NoSQL.	
Module -3	**8 Hours**
Fundamentals To Mongodb MongoDB: Introduction to MongoDB and its Features, Database, Collection and Documents, Data Types in MongoDB, Installation, The Mongo shell, CRUD Operations, Create Operations, Read Operations, Update Operations, Delete Operations Bulk Write, Aggregation, Aggregation Pipeline, Map-Reduce, Single Purpose Aggregation Operations	
Module -4	**8 Hours**
Working With Mongodb Text-search Text Indexes, Text Search Operation, Text search in the Aggregation Pipeline, Geospatial Queries, Find Restaurants with Geospatial Queries, Geo JSON Objects, Indexes-Single Field Indexes, Compound Indexes, Storage, Storage Engines, Journaling, GridFS, Replication: Replica Set Members, Replica Set Deployment Architecture, Replica Set High Availability, Replica Set Read and Write Semantics.	

Module -5	**8 Hours**
Building Web Applications Sharding: Sharded Cluster Components, Sharding strategy, Data Partitioning with Chunks, Sharded Cluster balancer, Web Application with MongoDB using PHP/Python/PyMongo, Connecting PHP/Python/PyMongo with MongoDB, CRUD Operations, Aggregate functions with PHP/Python /PyMongo	

Question paper pattern:

- There will be 2 full questions from each module.
- Each full question consists of 20 marks.
- Students have to answer 5 full questions selecting ONE from each module.

Text Books:

Kristina Chodorow: MongoDB: The Definitive Guide: Powerful and Scalable Data Storage, 2nd Edition, O'Reilly, 2013

Reference Books:

- Kristina Chodorow: MongoDB: The Definitive Guide: Powerful and Scalable Data Storage, 2nd Edition, O'Reilly, 2013.

Course Outcomes (CO):

CO1:Understand the significance of different databases

CO2:Design and develop queries for CRUD operations

CO3: Design and develop queries using Aggregation Framework and Pipeline to access MongoDB

CO4:Develop PHP/Python/Mongo programs for CRUD operations and Aggregation functions

COs	**Mapping with POs**
CO1	PO3
CO2	PO1,PO2,PO3,PO4, PO5,PO7,PO11
CO3	PO1,PO2,PO3, PO4, PO5,PO7,PO11
CO4	PO1,PO2,PO3,PO4, PO5,PO7,PO8,PO11,PO12

SEMESTER – III

Artificial Intelligence

Subject Code	**18MCA353**	CIE Marks	**50**
Number of Lecture Hours/Week	**3**	SEE Marks	**50**
Total Number of Lecture Hours	**39**	SEE Hours	**03**

CREDITS – 3:0:0

Course Learning Objectives (CLO):

- Identify the differences between knowledge representation and knowledge organization.
- Understand about Robotics and structure
- Learn different search strategies and fuzzy methodology
- Representation of knowledge and reasoning
- Evaluate knowledge about planning and learning strategies

Modules	**Teaching Hours**
Module -1	**7 Hours**
Artificial Intelligence Introduction: over view of AI, Importance of AI, AI- History, AI and related fields, search control strategy: preliminary concepts, AI-Applications, Turing Test, Application areas, Problem Space, Problem Characteristics, and AI Problems.	
Module -2	**8 Hours**
Fundamentals Of Robotics Robot anatomy-Definition, law of robotics, History and Terminology of Robotics-Accuracy and repeatability of Robotics-Simple problems-Specifications of Robot-Speed of Robot-Robot joints and links-Robot classifications-Architecture of robotic systems-Robot Drive systems	
Module -3	**8 Hours**
Fuzzy Logic Introduction, fuzzy set theory, classical sets, membership function, fuzzy rule generation, compliment, Intersections, Unions, combinations of operations, Aggregation operations. Fuzzy number, Linguistic variables, arithmetic operations on intervals and numbers, lattice of fuzzy numbers, fuzzy equations, fuzzy relations, fuzzy projections, fuzzy systems, fuzzy propositions, fuzzy inference, fuzzyfications, and defuzzification.	
Module -4	**8 Hours**
Probabilistic Reasoning Bayesian probabilistic inference, Bayes Theorem, Knowledge based system, representation of knowledge origination, knowledge	

manipulation	
Module -5	**8 Hours**
Planning And Natural Language Processing Introduction, Components of a Planning System, Goal Stack Planning, Hierarchical Planning. Linguistics, grammars and languages. Basic parsing techniques, expert system architecture, characteristics of expert systems, Rules for Knowledge in Language Understanding, Syntax and Semantic Analysis, NLP.	

Question paper pattern:

- There will be 2 full questions from each module.
- Each full question consists of 20 marks.
- Students have to answer 5 full questions selecting ONE from each module.

Text Books:

1."Artificial Intelligence-A modern Approach" Stuart Russel, Peter Norvig, second edition, PHI/Pearson Education, 2010
2. Artificial Intelligence – Structures and Strategies for Complex Problem Solving, George F. Luger, Pearson Education, 4/e, 2003.
3. Thimothi and Ross: Engineering Applications Fuzzy Fuzzy logic, PHI.

Reference Books:

1. Artificial Intelligence and Intelligent Systems, N. P. Padhy, Oxford Press, 4/e, 2008.
2. Artificial Intelligence: A new Synhesis Approach, Nils J. Nilson, Morgan Kaufmann, 1998.
3.G.J.Klir and B.Yuan: Fuzzy sets and Fuzzy logic, PHI, 1995

Course Outcomes(CO):

CO 1: Understand of Artificial intelligence concepts.
CO 2: Apply different Search strategies in problem solving
CO 3: Discover knowledge and perform reasoning.
CO 4: Derive planning strategies and machine learning techniques

Cos	Mapping with Pos
CO1	PO1, PO2, PO5, P10
CO2	PO1, PO2
CO3	PO1, PO2, PO4, PO5
CO4	PO1, PO2, PO5

SEMESTER –III

Network Architecture and Management Laboratory

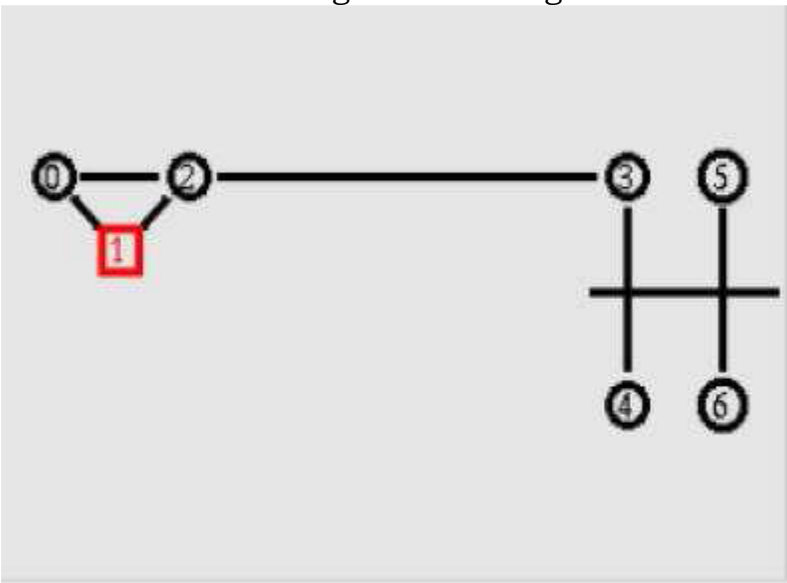
Laboratory Code	**18MCAL36**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**02**	**SEE Marks**	**50**
Total Number of Lecture Hours	**26**	**SEE Hours**	**03**

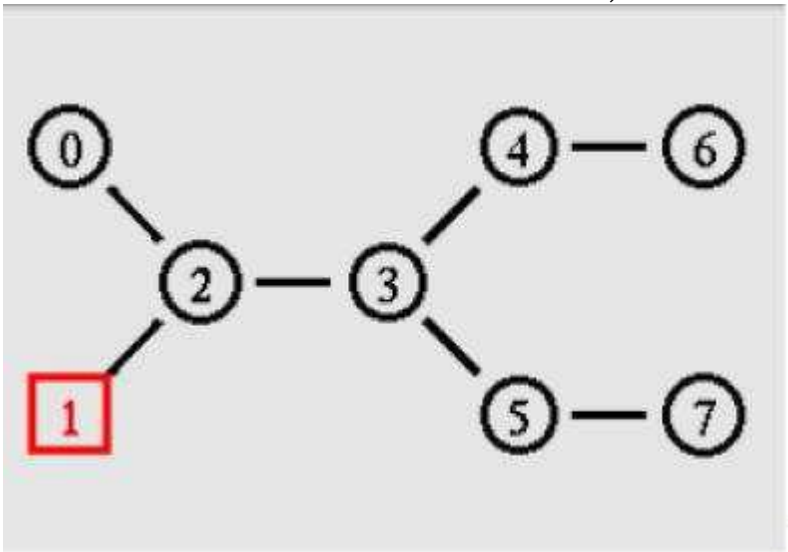
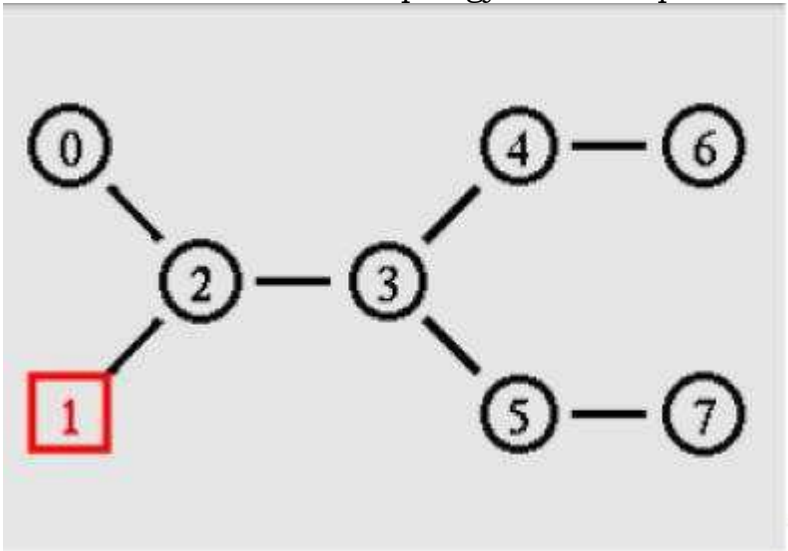
CREDITS – 0:0:1

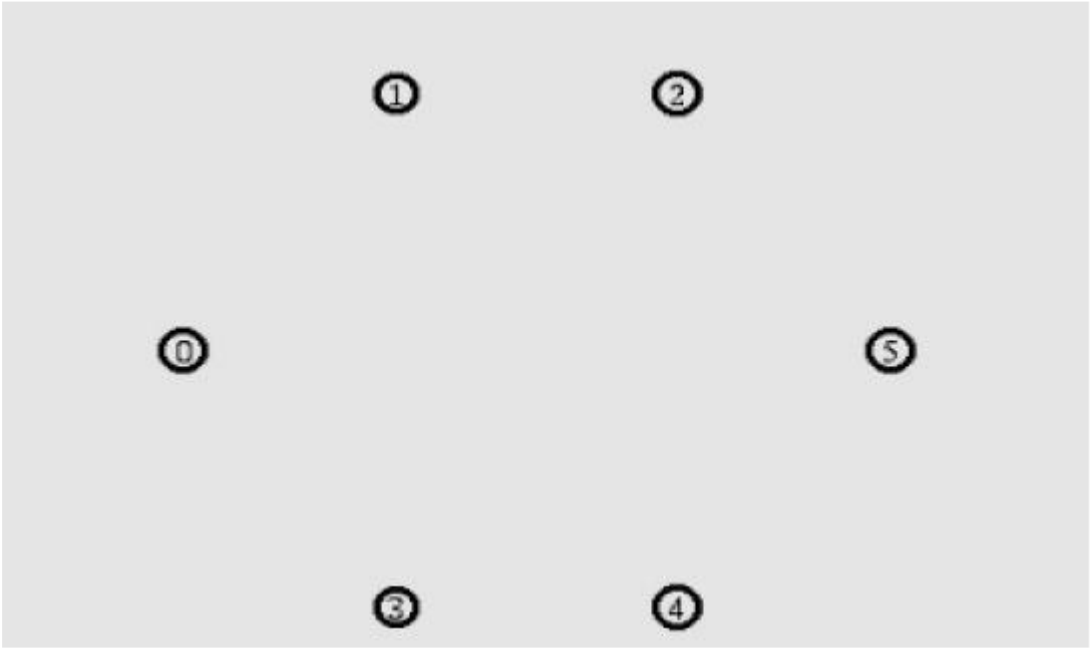
Course Learning Objectives (CLO):

- Demonstrate the architecture of networks and the flow of packets
- Analyze the significance of network management and apply the security tactics

1	**Write a TCL script to simulate the network described below** Consider a small network with five nodes n0, n1, n2, n3, n4, forming a star topology. The node n4 is at the center. Node n0 is a TCP source, which transmits packets to node n3 (a TCP sink) through the node n4. Node n1 is another traffic source, and sends UDP packets to node n2 through n4. The duration of the simulation time is 10 seconds. Write a Tcl script (in the text area below) to simulate this scenario. Once you have written your simulation script, click on the "Run" button to execute it. You can verify your code by clicking on the "View Solution" button at the bottom of the page
2	**Write a TCL script to simulate a file transfer with ns2** Consider a client and a server. The server is running a FTP application (over TCP). The client sends a request to download a file of size 10 MB from the server. Write a script to simulate this scenario. Let node #0 be the server and node #1 be the client. TCP packet size is 1500 B. Assume typical values for other parameters. **Note:** This simulation require transfer of a **fixed** size file. Therefore, time required for the transfer would be constant for a given bandwidth of a link. To verify this, determine the time that would roughly be required for the transfer. Then look at the bottom of the trace file and verify whether there is any transmission beyond the time calculated. To verify that the client has downloaded the entire file, plot the "Bytes Received " curve for node #1. The y-axis is in Kbits. Convert it to MB and verify whether it approximates the specified file size. TCP headers would effectively increase the count of received bytes at node # 1.
3	**Setting up a local area network with ns2** In this exercise you will be simulating a CSMA/CD based LAN with ns2. Consider the LAN with seven nodes to be an isolated one i.e. not connected to the Internet. Node # 0 in the LAN act as a UDP traffic source, and node # 6 is the destination node. Assume CBR traffic to be flowing between the nodes. The simulation lasts for 25

	seconds. In Ethernet a packet is broadcasted in the shared medium, and only the destination node accepts the packet. Other nodes simply drop it. What should be the number of hops a packet from node # 0 to node # 6 travel? Verify this from the "Hop Count" plot. **Additional Task:** Suppose the above LAN is to be connected to the Internet. Add node # 7 into the network so that it act as the gateway. Connect node # 0 and # 7 with a 1 Mb wired link. Move the UDP source to node # 7. How the hop count should get affected in this case? Verify from the plot.
4	**Simulating link errors** Consider the following network diagram  Here node # 2 act as a router. Any traffic to or from the LAN passes through it. Consider node # 1 running a FTP server, and node # 5 is downloading a file of size 4 MB. However, the link between node # 2 and # 3 is fault. It drops packets with a fixed probability of 0.2. Implement a link error model to reflect this. It may be noted here that the file download time will be more than the we had in exercise # 2 of experiment # 1. Try different values of the simulation time to ensure that the file has been entirely transferred. Is the plot of bytes received a linear curve or non-linear? Why? Presence of link errors cause one or more packets to be retransmitted. Verify this from the "Packet Retransmissions" plot. .
5	**Measuring Network Performances** **Bottleneck in the network** Consider a dumbbell topology with eight nodes as shown as in the following figure. Consider nodes# 2 and 3 to be two routers connecting two different networks. When the bandwidth of the link 2-3 is much lower than the sum of bandwidths of the

	other links in the network, it act as a bottleneck.  Assume node # 0 running a FTP application (over TCP) and sending data to node # 6. Node # 1 is sending CBR data node # 7. Assume all the links except 2-3 has a bandwidth of 1 Mb, propagation delay of 10ms and queue type as DropTail. (All are duplex links). **Tasks:** • The link 2-3 has a propagation delay of 10 ms. Vary it's bandwidth from 0.5 Mb to 2.5 Mb in steps of 0.25Mb. • Compute the throughput for node # 3 in each case • Plot the throughput vs. bandwidth data in the "Custom Plot" section below Based on the above plots, suggest what should be the recommended bandwidth of the link 2- 3. Now, plot the end-to-end delay between nodes 0 and 6 for the above chosen values of link 2-3 bandwidth. Revisit your previous answer (i.e. optimum bandwidth of link 2-3) based on these graphs.
6	**Bandwidth sharing between TCP and UDP** Consider the dumbbell topology from our previous exercise: 

	Node # 0 is a TCP source, and the corresponding sink is at node # 6. Node # 1 is a UDP source (CBR traffic) with a null agent attached to node # 7. These two traffic flows through the common link 2-3. The aim of this exercise is to examine how TCP and UDP share the bandwidth between themselves when the rate of CBR traffic is changed. Set the TCP packet size to 1460 B. The UDP and CBR packet sizes are 1500 B. All the links in the network have same bandwidths (say, 4 Mb), delay and queue types. **Part 1:** • Set the initial rate of CBR traffic to 0.5 Mb. Run the simulation, and plot the "Bytes Received" by node #s 4 and 5 (sinks for TCP and UDP traffic) • Now, increment the rate up to 4 Mb, the link bandwidth, in steps of 0.5 Mb. Run the simulation and plot the graphs again. How does the graphs change after each run? In particular, what's the nature of the graphs when the rate of CBR traffic is 50% of the bandwidth? **Part 2: Behaviour of UDP** • Reduce the bandwidth of the link 2-3 to say, 2 Mb. Repeat the above steps and observe the graphs in this case. From the graphs plotted observe how UDP occupies a larger portion of the bandwidth. How does the behaviour change for other variations of TCP (Newreno, Vegas)?
7	**Write a TCL script to simulate the following scenario with ns2 simulator.** Consider six nodes, (as shown in the figure below) moving within a flat topology of 700m x 700m.  The initial positions of nodes are 0 (150,300) ,1 (300,500),2 (500,500),3 (300,100),4(500,100) and 5(650,300) respectively.A TCP connection is initiated between node 0 (source) and node 5 (destination) through node 3 and node 4 i.e the route is 0-3-4-5. At time t = 3 seconds the FTP application runs over it. After time t=4.0 sec, node 3 (300,100) moves towards node 1 (300,500)

	with a speed of 5.0m/sec and after some time the path break, then the data transmit with a new path via node 1 and node 2 i.e the new route 0-1-2-5. The simulation lasts for 60 secs. In the above said case both the route has equal cost. Use DSR as the routing protocol and the IEEE 802.11 MAC protocol. Now Analyze the trace file and determine when the use of second route commence, and Plot the number of packets received by each node over the entire time duration of the simulation
8	**Simulate a wired network and demonstrate Distance Vector Routing algorithm .**

Course Outcomes:

CO: Design networks to implement network topologies, routing techniques and analyze the network performance under various networking conditions

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO	PO1,PO3,PO5,PO7,PO11

III SEMESTER Python Programming Laboratory			
Laboratory Code	**18MCAL37**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**02**	**SEE Marks**	**50**
Total Number of Lecture Hours	**26**	**SEE Hours**	**03**

CREDITS – 0:0:1

Course Learning Objective(CLO) :

- Learn basics concepts of python programming.
- Implement advanced programs in python based on the knowledge gained.

	Program Statements
1.	a) Write a Python program to compute the sum of first n given prime numbers. b)Write a program to compute 1/2+2/3+3/4+...+n/n+1 with a given n input by console (n>0).
2.	a)Python program to check withdrawal is possible or not, based on balance amount and withdrawal amount input from the user. b) Write a python program compress and decompress the string.
3.	a)Write a Python program to display some information about the OS where the script is running. b)Convert a string to camel case. E.g.: If the given string is "This is a test", the output should be "ThisIsATest" Find the number of vowels, consonants and digits in a given string.
4.	Write the python code to perform the list operations
5.	Write the python code to perform the Set operations
6.	a) Write a program to multiply two matrices using comprehension. b) Write a Python program to copy the contents of list of lists to tuples of tuple.
7.	Read a string from keyboard input. Create a list containing tuples, where each tuple represents a word in the input string and length of that string. Write a program sort the words in descending order of their length.
8.	Do the following using regular expressions:- • Find all occurrences of a word in a multiline string. The search must be case insensitive. Also find and display the starting index of each matched word in the input string. • Given a line of text find all characters other than vowels and space characters. • Validate USN using regular expressions. (both UG and PG) of VTU students. (Hint: UG USN format: 1RN15EC001, PG USN format: 1RN15MCA01)
9.	Do the following using regular expressions:-

	Find all occurrences of a word in a multiline string. • The search must be case insensitive. • Also find and display the starting index of each matched word in the input string. • Given a line of text find all characters other than vowels and space characters. Given a list of strings find all strings that start with a digit or an underscore.
10.	Create a dictionary for words and their meanings. Write functions to add a new entry (word: meaning) , search for a particular word and retrieve meaning, given meaning find words with same meaning , remove an entry, display all words sorted alphabetically. [Program must be menu driven.
11.	Write a Python Event-driven Program for file operations Press 1. Display first 5 lines and last 5 lines of a file. read mode 2. Display current position of the file pointer 3. Reposition the pointer at the beginning 4. Insert a line of text 5. Copy the contents of the whole file into another and display. 6. exit.
12.	Write an Object oriented Python program to demonstrate: i) Constructor, methods and objects of a class ii) Polymorphism iii) Inheritance iv) Implementing interfaces
13.	Write a python program to demonstrate connection to database and retrieve the information. Example: create menu driven program which will demonstrate add, display, modify and delete the record of an employee table created using database with attributes slno, name, address, empcode, dateofbirth, age, mobile, status, designation.
14.	Write a python program to demonstrate networking concepts.
Note 1: In the practical Examination each student has to pick one question from a lot of all the 14 questions.	
Course outcomes(CO): After completing the course the students are able to: **CO:** Design and develop an applications using Python Programing.	

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO	PO1,PO2,PO4,PO5,PO8,PO11

SEMESTER –III

Data Science Using R Laboratory

Laboratory Code	**18MCAL38**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**02**	**SEE Marks**	**50**
Total Number of Lecture Hours	**26**	**SEE Hours**	**03**

CREDITS – 0:0:1

Course Learning Objectives (CLO):

- To understand basic data analysis concepts in R
- To analyse different statistical methods in R
- To apply basic machine learning algorithms for different datasets
- To visualize data using different plots

Note: Students have to pick one from Part A and one from Part B.

Install R studio on windows and load required packages. (prerequisite)

PART A

1	Create Vector & List. Perform manipulation of list elements.
2	Create two matrices and implement matrix computations on them.
3	Write R script to import different types of data sets (.csv, .txt, .xlsx) and write modified datasets to specific location.
4	Create a data frame for student with Name, USN and Marks. Add row/column, Retrieve elements of columns and rows in the data frame.
5	Load a built in data set and view summary of individual variables. Perform different operations on data sets and export the data to csv file format.
6	Find the data distribution using box and scatter plot. Plot different types of charts on sample data.

PART B

1	Write R script to treat missing values.
2	Analyse measures of Central tendency for sample dataset. Compute covariance and infer correlation between the variables.
3	Formulate linear regression model for any data set and identify explanatory variables.
4	Write a R script to analyse the performance of k-means clustering and visualize the performance.
5	Write an R script to analyse KNN classifier and verify the performance using confusion matrix.

Course Outcomes:
CO: Apply different statistical, machine learning algorithms and visualize using R.

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO-1	PO5, PO11

Dr. Ambedkar Institute of Technology, Bengluru-56

(An Autonomous Institute, Approved by AICTE, Affiliated to V T U, Belagavi)
Nationally Accredited by NAAC with 'A' Grade

Master of Computer Applications

(Accredited by National Board of Accreditation)

MCA IV Semester Syllabus

2018-21 Scheme

		IV Semester							
Subject Code	Name of the Subject	Credits				Exam Duration in Hours	Marks		Total Marks
		Lecture	Tutorial	practicals	Total Credits		CIE	SEE	
18MCA41	Enterprise Applications-1	3	-	-	3	3	50	50	100
18MCA42	Advanced Web Technologies	3	-	-	3	3	50	50	100
18MCA43	Analytical Skills And Building Professional Development	2	1	-	3	3	50	50	100
18MCA44	Elective-2	3	-	-	3	3	50	50	100
18MCA45	Elective-3	3	-	-	3	3	50	50	100
18MCA46	Research Methodology	2	-	--	2	3	50	50	100
18MCAL47	Enterprise Applications Lab	--	-	1	1	3	50	50	100
18MCAL48	Advanced Web Technologies Lab	--	-	1	1	3	50	50	100
18MCAM49	Mini Project using Android	--	1	1	2	3	50	50	100
Total		**16**	**2**	**3**	**21**	**27**	**450**	**450**	**900**

ELECTIVE-2									
Subject Code	**Name of the Subject**	**Credits**				**Exam Duration**	**Marks**		**Total Marks**
		L	**T**	**P**	**Credits**		**CIE**	**SEE**	
18MCA441	Information Security	3	-	--	3	3	50	50	100
18MCA442	Data Mining & Business Intelligence	3	-	--	3	3	50	50	100
18MCA443	Object oriented Modelling & Design	3	-	--	3	3	50	50	100

ELECTIVE-3									
Subject Code	**Name of the Subject**	**Credits**				**Exam Duration**	**Marks**		**Total Marks**
		L	**T**	**P**	**Credits**		**CIE**	**SEE**	
18MCA451	Software Quality & Performance Evaluation	3	-	--	3	3	50	50	100
18MCA452	Software Architecture	3	-	--	3	3	50	50	100
18MCA453	Enterprise Resource Planning	3	-	--	3	3	50	50	100

ENTERPRISE APPLICATIONS SEMESTER – IV			
Subject Code	18MCA41	**CIE Marks**	50
Number of Lecture Hours/Week	3	**SEE Marks**	50
Total Number of Lecture Hours	39	**SEE Hours**	03
CREDITS – 3:0:0			

Course Learning objectives(CLO):

- Present J2EE concepts and designing database access with java applications.
- Introduce the concepts of server side programming using Servlets & JSP.
- Present different types of enterprise java beans and implement them.
- Design and developing an application using spring and Hibernate frame work.

Modules	**Teaching Hours**
Module -1	
Annotations and JDBC Creating Packages, Interfaces, JAR files and Annotations. The core java API package, New java.Lang Sub package, Built-in Annotations with examples. The Concept of JDBC, JDBC Driver types, A brief overview of JDBC process, Database Connection, Statement objects, ResultSet, Transaction Processing, Metadata, Datatypes, Exceptions. Introduction to Embedded SQL with JDBC.	**07 Hours**
Module -2	**8 Hours**
Servlet and JSP and Controlling the Structure of generated servlets J2EE Architecture,Servlet Structure, Servlet packaging, HTML building utilities, Lifecycle, SingleThreadModel interface, Handling Client Request: Form Data, Handling Client Request: HTTP Request Headers. Generating server Response: HTTP Status codes, Generating server Response: HTTP Response Headers, Handling Cookies, Session Tracking.	
Module -3	**8 Hours**

JSP and Controlling the Structure of generated servlets Overview of JSP: JSP Technology, Need of JSP, Benefits of JSP, Advantages of JSP, Basic syntax. Invoking java code with JSP scripting elements, creating Template Text, Invoking java code from JSP, Limiting java code in JSP, using jsp expressions, comparing servlets and jsp, writing scriptlets. For example Using Scriptlets to make parts of jsp conditional, using declarations, declaration example. Controlling the Structure of generated servlets: The JSP page directive, JSP Action tags.	
Module - 4	**6 Hours**
Java Beans Working with Java Beans. Introspection, creating java bean, manifest file, Bean Jar file, adding controls, Bean properties, Simple properties, bound properties, Icon, Bean info class, Constrained Properties, Persistence, Java Beans API. **EJB and Server Side Component Models** Enterprise Java Beans: The EJB Container, EJB Classes, EJB Interface. Deployment Descriptor, Session Java Bean, Entity Java Bean, Message-Driven Bean, The JAR File.	
Module – 5	**10 Hours**
Spring Framework Introduction to Spring Framework, Spring Framework architecture, IOC-containers, Bean scopes, Bean Life cycle, Dependency Injection, Beans wiring, Event Handling in springs, Custom events in springs Spring AOP, and Spring JDBC. **Spring MVC** Spring MVC : Spring 3.0 features –Introduction to Spring MVC – Handler Mapping –Controllers –Validations –Handler Interceptors – Views –Form tags.	

Question paper pattern:

- There will be 10 questions with 2 full questions from each module.
- Each full question consists of 20 marks. Students have to answer 5 full questions, selecting ONE from each module.

Text Books:

1. Marty Hall, Larry Brown. Core Servlets and Java Server Pages. Volume 1: Core Technologies. 2nd Edition. (Chapter 3,4,5,6,7,8,9,10,11,12,13,14).
2. Java 6 Programming Black Book, Dreamtech Press. 2012 (Chapter 17,18,19,20,21,22,27,28,29,30).
3. Andrew LeeRubinger, Bill Burke. Developing Enterprise Java Components. Enterprise JavaBeans 3.1.O'reilly. (Chapter 1,2,3,4,5,6,7,8,9,10,11).

Reference Books:

1. Michael Sikora, EJB 3 Developer Guide, A practical guide for developers and architects to the Enterprise Java Beans Standard, Shroff Publishers & Distributors PVT LTD. July 2008.
2. Herbert Schildt, Java The Complete Reference, 8th Edition. Comprehensive coverage of the Java Language. Tata McGraw-Hill Edition – 2011.

Course Outcomes(CO):

CO1: Understand Concept of enterprise applications programming.
CO2: Analyze the methodologies and constraints of implementation.
CO3: Apply the enterprise programming methodologies to design applications.
CO4: Design and Develop applications to be deployed in real world scenarios.

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO1,PO2,PO4,PO5,PO8
CO 2	PO1,PO2,PO3,PO4
CO 3	PO1,PO2,PO3,PO4,PO5
CO 4	PO1,PO2,PO3,PO4,PO5,P11

Advanced Web Technologies
SEMESTER – IV

Subject Code	**18MCA42**	CIE Marks	**50**
Number of Lecture Hours/Week	**3**	SEE Marks	**50**
Total Number of Lecture Hours	**39**	SEE Hours	**03**

CREDITS – 3:0:0

Course Learning Objectives (CLO):

- To design web pages using Bootstrap framework
- To develop different approaches of Web technologies using PHP and Ruby on Rails.
- To design Single page web applications using AngularJs
- To design asynchronous web applications using Ajax.

Modules	**Teaching Hours**
Module -1	**8 Hours**
Bootstrap- Introduction, **Layout**: Container, Grid, **Components**: Alerts, Badge, Card, Jumbotron, Buttons/Buttons group, Navs/Navbar, Pagination, Modal, Carousel, Collapse, Form, Input group, Progress bar	
Module -2	**8 Hours**
Introduction to PHP-Origins and uses of PHP, Overview of PHP, Primitives, operations and expressions, Output, Control statements, Arrays, Functions, Pattern matching, Form handling, Files handlers. Building Web applications with PHP- Using databases, tracking users- cookies, sessions.	
Module -3	**8 Hours**
Ajax- AJAX principles, Creating Ajax applications, Adding Server-side programming, Sending data to the server using GET and POST. Downloading JavaScript, Connecting to Google suggest. Ajax Patterns-Periodic fetch, Periodic refresh, case study.	
Module -4	**8 Hours**
Angular JS -Single Page Applications: -Introduction, MVC Architecture, Data binding, binding with lists, Angular Directives, Controller, Dependencies, Bootstrapping an angular applications, scope and views.	
Module -5	**7 Hours**
NodeJs-Introduction, JavaScript closures, Node Modules-Common JS Modules-core modules, third-party modules, file modules, folder modules, Developing Node.js web application.	

Question paper pattern:

- There will be 2 full questions from each module.
- Each full question consists of 20 marks.
- Students have to answer 5 full questions selecting ONE from each module.

Text Books:

1. "Bootstrap Essentials",Snig Bhaumik, PACKT publishing
2. Robert W. Sebesta: Programming the World Wide Web, 7th Edition, Pearson Education,2012.
3. Steven Holzner: Ajax: A Beginner's Guide, Tata McGraw Hill, 2011
4. Amos Q. Haviv," MEAN Web Development", Packt Publishing,2014.

Reference Books:

1. Steven Holzner "Complete Reference-PHP ", Tata Mc Graw Hill,2008
2. "Bootstrap programming cook book"-Fabio Cimo,Web Code Geeks
3. Adam Bretz and Colin J. Ihrig," Full Stack JavaScript Development with MEAN", SitePoint Pty. Ltd.,2014.
4. Nicholas C Zakas et al: Professional AJAX, Wiley India, publications

Course Outcome (CO):

CO1: Design web applications using Responsive designs.

CO2: Design and develop web applications using PHP and MYSQL.

CO3: Design Single page web applications using AngularJs and Node JS.

CO4: Design asynchronous web applications using Ajax.

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO1	PO5,PO11
CO2	PO2,PO4,PO5,PO11
CO3	PO2,PO4,PO5,PO7,PO11
CO4	PO5,PO11

Analytical Skills And Building Professional Development SEMESTER IV			
Subject Code	**18MCA43**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**3**	**SEE Marks**	**50**
Total Number of Lecture Hours	**39**	**SEE Hours**	**3**
CREDITS – 2:1:0			

Course Learning Objectives(CLO)

- To solve basic problems on number systems.
- To implement time and work problem, data interpretation.
- To analyze and distinguish between reasoning, Verbal analogies, verbal classification.
- Apply Motivation principles & demonstrate communication skills

Modules	**Teaching Hours**
Module -1*: Quantitative Aptitude*	**8 Hours**
Number Systems, Profit, Loss and Discount, LCM and HCF, Speed, Time and Distance, Percentages, Time and Work, Ratio and Proportion, Averages, Simple and Compound Interest, Sequences and Series, Inequalities, Probability, Surds and Indices, Permutation and Combination, Coordinate Geometry, Geometry, ,Mensuration,	
Module -2: Data Interpretation & Logical Reasoning	**8 Hours**
Data Handling, statistics, Venn Diagrams, Basic Reasoning Questions, Calendars, Clocks, Number and Letter Series, Seating Arrangement, Logical Matching, Logical Sequence, Syllogism, Logical Connectives, Blood Relations, statistics. Cracking corporate Real time quantitative test.	
Module -3	**8 Hours**
Improving the quality of Reading, Writing, Speaking and Listening to satisfy international academic requirements. Assignments and Online assessments	
Module -4	**7 Hours**
Motivation Skills & Personality Development, Goal Setting, Career Planning, Resume Building, Psychometric Test, Priority Management & Time Management, Positive Attitude and Self Confidence.	
Module -5	**7 Hours**
Technical Writing and Professional Etiquette: Building project proposals, brochures, newsletters, articles. Professional Etiquettes: Social and Professional etiquette, email etiquettes	

Question paper pattern:

- The question paper will have five questions. All questions are compulsory.

Module 2 and 3 shall have internal choice.

- Each full question consists of 20 marks.
- Questions are set covering all the topics under each module.

1. R.S. Aggarwal, Quantitative Aptitude, S. Chand & Company, New Delhi, 2012. 2.
2. Govind Prasad Singh and Rakesh Kumar, Text Book of Quickest Mathematics (for all Competitive Examinations), Kiran Prakashan, 2012.
3. John Hoover & Angelo Valenti, "Unleashing Leadership", Jaico publishing House –WWW.JAICOBOOKS.COM
4. Kundu, C.L – "Personality development", Sterling Bangalore. 5. Sandra D. Collins, "Listening and Responding", Cengage Learning India, 2 nd Edition, 2008.

Reference Books:

1. R.S. Aggarwal, Objective Arithmetic, S. Chand & Company, New Delhi, 2005. 4. Dr. Lal, Jain,
2. Dr. K. C. Vashistha, "U.G.C.- NET/JRF/SET Teaching & Research Aptitude", Upkar Prakashan, 2010.
3. Wallace: "Personality Development", 1st Edition, 2008 Cengage Learning India.
4. Richard Denny, "Succeed for your self", Kogan page India, 3rd Edition. www.vivagroupindia.com.

Course Outcomes:

CO1:Demonstrate number system and probability

CO2: Apply time , work and data interpretation to solve real world problem

CO3: Apply logical, analytical reasoning and verbal analogies

CO4: Apply Motivation principles & demonstrate communication skills

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO1,PO2
CO 2	PO1,PO2
CO 3	PO1,PO2
CO 4	PO6,PO7,PO9,

Information Security SEMESTER –IV			
Subject Code	**18MCA441**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**3**	**SEE Marks**	**50**
Total Number of Lecture Hours	**39**	**SEE Hours**	**03**
CREDITS – 3:0:0			

Course Learning Objectives (CLO)

- Introduce students to the area of Information Security, cybercrime and Forensics.
- To understand the motive and causes for cybercrime, detection and handling.
- To analyze the areas affected by cybercrime and investigation tools used in cyber forensic.
- To Evaluate the knowledge of report writing and Forensic ethics

Modules

Module -1	**8 Hours**
INTRODUCTION: The Security Problem in Computing: The meaning of Information Security and computer Security, Computer Criminals, Methods of Defense, and Elementary Cryptography: Substitution Ciphers, Transpositions, Making "Good" Encryption algorithms, The Data Encryption Standard, The AES Encryption Algorithms, Public Key Encryptions, and Uses of Encryption.	
Module -2	**6 Hours**
Program Security: Sphere, Terminology, Vulnerability in the Cyber Structure and Infrastructure, Cyber threats and Weaponry, Cyber Defense, Cyber Defense, Cyber Attack Detection and Prevention, Information Security Testing, Cyber Security Investigation/assessment, Cyber Deterrence.	
Module -3	**8 Hours**
Cyber Crimes and Cyber Laws: Classification of Cyber Crimes, IT laws & Cyber Crimes-Internet, Hacking, Password Cracking, Viruses, Virus Attacks, Reasons for Commission of Cyber Crimes, Malware and	

its type Adware, Spyware, Browser hijacking software, Virus, Worms, Trojan Horse, Scareware, Kinds of Cyber Crime.	
Module -4	**10 Hours**
Administrative Security: Security planning, Risk Analysis, Organizational Security policies, Physical Security, Legal Privacy and Ethical Issues in computer security: Protecting programs and data, Information and the law, Rights of Employees and Employers, Software failures, Computer crime, Ethical Issues in computer security.	
Module -5	**7 Hours**
Database Security: Security requirements, Reliability and integrity, Sensitivity data, proposals for multilevel security. Security in network: Threats in network, network security controls, Firewalls, Secure E-Mail.	
Question paper pattern: • The question paper will have ten questions. Choose one Question from each module • Each full question consists of 20 marks. • Questions are set covering all the topics under each module.	

Text Books:

1. SunitBelapure and Nina Godbole, " Cyber Security: Understandign Cyber Crimes, Computer Forensics And Legal Perspectives", Wiley India Pvt Ltd, ISBN:978-81-265-2179-1. Publish Date 2013.
2. Surya prakash Tripathi, Rajendra Goel, and Praveen Kumar Shukla, "Introduction to Information Security and Cyber Laws", DT Editorial Services

Reference Books:

1. Thomas J. Mowbray, " Cybersecurity: Managing Systems, Conducting Testing, and Investigating Instrusions", Copyright@2014 by John Wiley & Sons, Inc. ISBN:978-1-118-84965-1.2014.

2. James Graham. Ryan Olson. Rick Howard. "Cyber Security Essentials", CRC Press. 15-Dec-2010.

Course Outcome (CO): At the end of this course, the students will be able to

CO1.Describe knowledge on the Information Security and cyber security, cybercrime and forensics.

CO2: Understand a tools and methods used in cybercrime and know about the tools and techniques for the forensics.
CO3: Ability to apply strict policies and procedures with meticulous record keeping.
CO4: Derive the digital forensics

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO1	PO1,PO2,PO5,PO10
CO2	PO1,PO2
CO3	PO1,PO2,PO4,PO5
CO4	PO1,PO2,PO5

Data Mining & Business Intelligence SEMESTER – IV			
Subject Code	**18MCA442**	CIE Marks	**50**
Number of Lecture Hours/Week	**3**	SEE Marks	**50**
Total Number of Lecture Hours	**39**	SEE Hours	**03**
CREDITS – 3:0:0			

Course Learning Objectives (CLO):

- Learn about Business Intelligent and Decision Support system
- Understand the concept of Data warehousing and OLAP, storage and retrieval technique of data from DATA CUBE
- Analyze different types of data and different preprocessing techniques
- Analyze different data mining techniques(Association, Classification & Clustering)

Modules	Teaching Hours
Module -1	**8 Hours**
BUSINESS VIEW OF INFORMATION TECHNOLOGY APPLICATIONS Business Enterprise, Functions and Core Business Processes, Baldrige Business Excellence Framework, Purpose of Using IT in Business, Application development Approaches, Information Users and Their Requirements, Types of Digital Data: Structured Data, Unstructured Data, Semi-Structured Data, Characteristics, Issues and Challenges	
Module -2	**8 Hours**
BUSINESS INTELLIGENCE BI Overview, BI Skill requirements, BI benefits, functions and Applications, Using Analytical Information for Decision Support, Role of DSS, EIS, MIS, Business Analytics, BI Component Framework, BI Users, Applications, Popular BI Tools	
Module -3	**8 Hours**
INTRODUCTION TO OLTP AND OLAP: Characteristics, Issues and Challenges, Comparison, Dimensional Data, Different OLAP Architectures, ROLAP, MOLAP, HOLAP, Data Models for OLTP and OLAP, OLAP operations **Data Mining And Its Applications** Introduction, What is Data Mining, Motivating Challenges, Data Mining Tasks, Which technologies are used, which kinds of applications are targeted by Data Mining , Types of Data, Data Mining Applications, Data Preprocessing	
Module -4	**8 Hours**
ASSOCIATION ANALYSIS: BASIC CONCEPTS AND ALGORITHMS Frequent Item set Generation, Rule Generation, Compact Representation of Frequent Item sets, Alternative methods for generating Frequent Item sets, FP Growth Algorithm	

Classification: Methods, Improving Accuracy Of Classification Basics, General approach to solve classification problem, Decision Trees, Rule Based Classifiers, and Nearest Neighbor Classifiers. Bayesian Classifiers	
Module -5	**8 Hours**
CLUSTERING TECHNIQUES AND OUTLIER ANALYSIS Overview, Features of cluster analysis, Types of Data and Computing Distance, Types of Cluster Analysis Methods, Partitional Methods, Hierarchical Methods, Density Based Methods Outlier detection methods	

Question paper pattern:

- There will be 2 full questions from each module.
- Each full question consists of 20 marks.
- Students have to answer 5 full questions selecting ONE from each module.

Text Books:

1. R N Prasad, Seema Acharya, "Fundamentals of Business Analytics" , Wiley India, 2011.
2. Jiawei Han and Micheline amber: Data Mining - Concepts and Techniques, 2nd Edition, Morgan Kaufmann Publisher, 2006.
3. Pang-Ning Tan, Michael Steinbach, Vipin Kumar: Introduction to Data Mining, Addison- Wesley, 2005.

Reference Books:

1. Arun K Pujari: Data Mining Techniques University Press, 2nd Edition, 2009.
2. G. K. Gupta: Introduction to Data Mining with Case Studies, 3rd Edition, PHI, New Delhi, 2009.
3. Alex Berson and Stephen J.Smith: Data Warehousing, Data Mining, and OLAP Computing McGrawHill Publisher, 1997.

Course Outcomes (CO):

CO 1: Understand the basic concepts of Business Intelligent and Decision Support system, Data Warehousing and OLAP & Data cube implementation.
CO 2: Illustrate Data Mining Challenges, applications & Pre-processing techniques.
CO 3: Analyse algorithms for Associations Analysis.
CO 4: Demonstrate different Classification techniques.

OBJECT-ORIENTED MODELING AND DESIGN PATTERNS SEMESTER IV			
Subject Code	**18MCA443**	CIE Marks	**50**
Number of Lecture Hours/Week	**3**	SEE Marks	**50**
Total Number of Lecture Hours	**39**	SEE Hours	**03**
CREDITS – 3:0:0			

Course Learning Objectives(CLO) :

- To apply the object oriented concepts for designing object oriented models.
- To analyze and design the software models using UML notations.
- To analyze the domain, application artifacts, and construct domain model and application model.
- To design class models using forward and reverse engineering concepts.
- To understand patterns and implement design patterns to provide solutions to real world software design problems.

Modules-1	**Teaching Hours**
Modeling Concepts & Class Modeling : What is OO development? OO themes, OO modeling history. Modeling as Design Technique: Modeling; abstraction, The three models. History of UML Building Blocks. **Object and class concepts**: Link and associations concepts, Generalization and inheritance, A sample class model, **Advanced object and class concepts:** Association ends; N-array associations; Aggregation; Abstract classes; Multiple inheritance; Metadata; Reification; Constraints; Derived data; Packages.	**10 Hours**
Module -2	**6 Hours**
State Modeling and Interaction Modeling: State Modeling: Events, States, Transitions and Conditions; State diagrams; State diagram behavior. **Advanced State Modeling:** Nested state diagrams; Nested states; Signal generalization; Concurrency; A sample state model; **Interaction Modeling:** Use case models; Sequence models; Activity models. Use case relationships; Procedural sequence models;	
Module -3	**10 Hours**
System Conception and Analysis: **System Conception**: Devising a system concept; elaborating a concept; preparing a problem statement. **Overview of analysis**: Domain class model; Domain state model; Domain interaction model; Iterating the analysis. **Application Analysis**: Application interaction model; Application class model; Application state model; adding operations. **Concept of Forward Engineering and Reverse Engineering:** Forward Engineering and Reverse Engineering of all diagrams of UML 2.0. Reverse engineering; Building the class models; Building the interaction model; Reverse engineering tips.	
Module -4	**5 Hours**

Pattern Introduction: What is a pattern and what makes a pattern? Pattern categories; Relationships between patterns; Pattern description. Idioms: Introduction; what can idioms provide? Idioms and style; Where to find idioms; Singleton, Counted Pointer example.	
Module -5	**8 Hours**
Design Patterns : Introduction, Model View Controller, Structural decomposition- Whole-Part, Access Control- Proxy; Creational Patterns – Factory Method, Singleton Structural Patterns – Adapter, Management Patterns- Command processor, Publisher-Subscriber	

Question paper pattern:

- There will be 10 questions with 2 full questions from each module.
- Each full question consists of 20 marks.
- Students have to answer 5 full questions, selecting ONE from each module.

Text Books:

5. Michael Blaha, James Rumbaugh, "Object-Oriented Modeling and Design with UML", 2nd Edition, Pearson Education / PHI, 2007
6. Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal, "Pattern-Oriented Software Architecture, A System of Patterns", Volume 1, John Wiley and Sons, 2006.
7. Booch, G., Rumbaugh, J., and Jacobson, I., "The Unified Modeling Language User Guide", 2nd Edition, Pearson, 2005.

Reference Books:

1. Grady Booch et al, "Object-Oriented Analysis and Design with Applications", 3rd Edition, Pearson, 2007.
2. Mark Priestley, "Practical Object-Oriented Design with UML", 2nd Edition, Tata McGraw-Hill, 2003.
3. Michael R Blaha, James R Rumbaugh, "Object Oriented Modeling and Design with UML", 2nd Edition, Prentice Hall, 2004.

Course Outcomes (CO):

CO1**:** Demonstrate the ability to apply the knowledge of object oriented concepts for designing system models.

CO2: Design and implement object oriented models using UML appropriate notations.

CO3: Apply the concept of domain and application analysis for designing Domain and application models, Build class models using forward and reverse engineering.

CO4: Implement patterns for constructing software designs of real world problems.

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO1,PO3
CO 2	PO1
CO 3	PO3,PO8
CO 4	PO3,PO4,PO8

Software Quality & Performance Evaluation

SEMESTER –IV

Subject Code	**18MCA451**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**3**	**SEE Marks**	**50**
Total Number of Lecture Hours	**39**	**SEE Hours**	**03**

CREDITS – 3:0:0

Course Learning Objectives (CLO)

- Able to know about the system performance, common mistakes in performance evaluation.
- Evaluation techniques, Performance metrics, performance requirements.
- Types of workload, characterization of workload techniques.
- Simulation common mistakes, types, and verification techniques.

Modules

Module -1	**4 Hours**
Introduction Outline of topics, Art of Performance evaluation **Common Mistakes and How to Avoid Them** Common Mistake in Performance Evaluation, Systematic approach to Performance Evaluation	
Module -2	**5 Hours**
Selection of Techniques and Metrics Selecting the evaluation Technique, Selecting the performance metrics, Commonly used Performance metrics, Utility classification of Performance metrics, setting performance requirements.	
Module -3	**10 Hours**
Types of Workloads Addition Instructions, Instruction Mixes, Kernels, synthetic Programs, Application Bench Marks, Popular Benchmarks, **Workload Characterization Techniques** Terminology, Averaging, Specifying Dispersion, Single parameter histogram, Multi parameter histogram, Principle Component Analysis, Markov Model, Clustering.	
Module -4	**10 Hours**
Introduction to simulation Common mistake in simulation, Other causes of simulation analysis failure, Terminology, selecting a language for simulation, Types of Simulation, Event set algorithms.	

Module -5	**10 Hours**
Model Verification Techniques Top down modular design, Anti bugging, Structured walk through, Deterministic models, run simplified cases, Trace, online display, Continuity test, Degeneracy test, consistency test, seed independence	
Question paper pattern: • There will be 10 questions with 2 full questions from each module. • Each full question consists of 20 marks. • Students have to answer 5 full questions, selecting ONE from each module.	
Text Books: 1. Raj Jian. "The Art of Computer Systems Performance Analysis". John Wiley and sons, New York, USA, 1991	
Reference Books: 1.Trivedi K S, **"**Probability and Statistics with Reliability, Queuing and Computer Science Applications", Prentice Hall of India, Reprinted in 1990 2. Law A M and Kelton W.D. "Simulation Modeling and Analysis ", McGraw Hill, New York,USA, 1991	
Course Outcome (CO): • CO1: Recognize the system performance, common mistakes in performance evaluation. • CO2: Demonstrate the Evaluation techniques, Performance metrics, performance requirement • CO3: Analyze the work load and characterization. • CO4: Evaluate the fundamentals of system simulation and model verification.	

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO1	**PO9**
CO2	**PO11**
CO3	**PO3, PO12**
CO4	**PO6,PO11**

ENTERPRISE RESOURCE PLANNING SEMESTER – IV			
Subject Code	18MCA452	**CIE Marks**	50
Number of Lecture Hours/Week	3	**SEE Marks**	50
Total Number of Lecture Hours	39	**SEE Hours**	03
CREDITS – 3:0:0			

Course Learning Objectives (CLO):

- Analyze a business' enterprise activities, workflow and process to identify problems, weaknesses, strengths, threats, opportunities,
- Stakeholders and entities interacting with the enterprise. Propose reengineered enterprise processes that optimize the
- Enterprise's performance.
- Evaluate ERP Products and Enterprise solutions understand the ERP Implementation Methodologies Assess ERP Modules.

Modules	**Teaching Hours**
Module -1	**8 Hours**
Introduction ERP as Integrated Management Information System - Evolution of ERP - Benefits of ERP. ERP vs Traditional Information Systems.	
Module -2	**8 Hours**
BUSINESS PROCESS REENGINEERING Business Process Reengineering- need and challenges, - Management concerns about BPR. - BPR to build business Model for ERP. ERP & Competitive advantage, - Basic Constituents of ERP, Selection criteria for ERP Packages. Procurement process for ERP Package.	
Module - 3	**8 Hours**
ERP PACKAGES Overview of ERP packages – PEOPLE SOFT, SAP-R/3, BAAN IV, MFG/PRO, IFS/AVALON, ORACLE- FINANCIAL, Survey of Indian ERP Packages regarding their Coverage, performance & cost.	
Module -4	**8 Hours**
ERP IMPLEMENTATION ERP Implementation- issues, Role of Consultants, Vendors, Users, - Need for training, customization. ERP implementation methodology and post implementation issues and options.	
Module -5	**7 Hours**
ERP CASE STUDIES ERP Case Studies In HRM, Finance, Production, Product Database, Materials, Sales & Distribution.	

Question paper pattern:

- There will be 10 questions with 2 full questions from each module.
- Each full question consists of 20 marks.
- Students have to answer 5 full questions, selecting ONE from each module

Text Books:

1. Alexis Leon, "ERP Demystified", Tata McGraw Hill, 1999.
2. Bret Wagner, Ellen Monk, "Concepts in Enterprise Resource Planning", 2012.
3. Bret Wagner, Ellen Monk, "Enterprise Resource Planning", Third Edition Cengage Learning, 2008.

Reference Books:

1. Vinod Kumar Garg and N.K .Venkata Krishnan, "Enterprise Resource Planning concepts and Planning", Prentice Hall, 1998.
2. Jose Antonio Fernandz, " The SAP R /3 Hand book", Tata McGraw Hill
3. Ashu Gupta, Rajesh Verma, Jatindar kumar, "Enterprise Resource Planning: Concepts and Applications", 2012.

Course Outcome (CO):

CO1: Identify the evolution of ERP and various benefits

CO2: Demonstrate ERP Products and Enterprise solutions

CO3: Implement ERP package to a Business Enterprise

CO4: Analyze ERP Modules for Business Enterprises

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO 1	PO3, PO6
CO 2	PO9
CO 3	PO7
CO 4	P012

Software Architecture SEMESTER –IV			
Subject Code	**18MCA453**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**3**	**SEE Marks**	**50**
Total Number of Lecture Hours	**39**	**SEE Hours**	**03**
CREDITS – 3:0:0			

Course Learning Objectives(CLO):

1. Designed to understand emerging field of software architecture in software development and the critical need for the development of such architectures.
2. Develop ability to perform the analyses necessary to formulate effective software architectures.
3. Analyze Software Engineering problems in terms of architectural thinking.
4. Apply the architectural concepts of platform, framework, pattern in architecting applications

Modules	**Teaching Hours**
Module -1	**07 Hours**
Introduction and Context of Software Architecture. The Architecture Business Cycle. What software architecture is and what it is not; What makes a "good" architecture? What it is and what it is not? Other points of view. Architectural patterns, Reference Models and Reference Architectures. Importance. Architectural structures and views	
Module -2	**8 Hours**
Architecture and Quality Attributes: System Quality Attributes. Quality attribute scenarios in practice. System quality attributes. Achieving Quality attributes: Availability; Interoperability; Modifiability; Performance; Security; Testability; Usability, Business Qualities	
Module -3	**8 Hours**
Applying Tactics: Availability tactics, Interoperability tactics; Modifiability tactics; Performance tactics; Security tactics; Testability tactics; Usability tactics	
Module -4	**8 Hours**

Architectural styles: Pipes and Filters, Data Abstraction and OO organization, Event-based, Implicit Invocation, Layered Systems, Repositories, Interpreters, Process Control Architectural Patterns - I: From Mud to structure; Layers; Pipes and filters; Blackboard	
Module -5	**8 Hours**
Architectural Patterns – II: Distributed systems: Broker; Interactive systems: Model-view-control Case Studies: Keyword to context, Instrumentation Software, Mobile Robotics	

Question paper pattern:

- There will be 10 questions with 2 full questions from each module.
- Each full question consists of 20 marks.
- Students have to answer 5 full questions, selecting ONE from each module

Text Book:

- Len Bass, Paul Clements, Rick Kazman, "Software Architecture in Practice" 3rd Edition ISBN-13: 978-0321815736, ISBN-10: 9780321815736
- Frank Buschmann, Regine Meunier, et al "Pattern Oriented Software Architecture" 2nd Edition – John Wiley & Sons , ISBN-10: 8120310411

Reference Books:

Nick Rozanski, Eoin Woods "Software Systems Architecture working with stake holders using viewpoints and perspectives" I edition ISBN-published by Addison Wesley 13: 978-0321112293, ISBN-10: 0321112296

Course Outcome (CO):

CO 1: To acquire the knowledge of the context and importance of software architecture and quality maintenance

CO 2: To apply the knowledge of various architectural tactics in multiple scenarios to enhance software quality

CO 3: To comprehend an architectural style as patterns

CO 4: To analyse and apply architectural style in multiple contexts.

COs	**Mapping with PO**
CO1	PO1, PO3
CO2	PO1, PO3, PO8
CO3	PO1, PO5, PO8
CO4	PO1, PO3, PO8, PO10

RESEARCH METHODOLOGY Semester IV			
Subject Code	**18MCA46**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**2**	**SEE Marks**	**50**
Total Number of Lecture Hours	**26**	**SEE Hours**	**03**
CREDITS – 2:0:0			

Course Learning Objectives (CLO):

1. To have a basic understanding of the underlying principles of quantitative and qualitative research.
2. To identify the overall process of designing a research study from its inception to its report.
3. To choose the most appropriate research method to address a particular research question.
4. To gain overview of a range of quantitative and qualitative approaches to data analysis.
5. To learn to write research report.

Modules	Teaching Hours
Module -1 Overview of Research and its types, Research approaches, Significance of Research, Research Methods versus Methodology. Research Process. Criteria of Good Research. Identifying and Identifying research problem, Technique Involved in Defining a Problem.	**5 Hours**
Module -2	
Introduction to different research designs. Essential constituents of Literature Review. Basic principles of experimental design, Primary data and Secondary Data, methods of primary data collection, classification of secondary data, designing questionnaires and schedules.	**5 Hours**
Module -3	
Sampling Methods Probability sampling: simple random sampling, systematic sampling, stratified sampling, cluster sampling and multistage sampling. Nonprobability sampling: convenience sampling, judgment sampling, quota sampling. Sampling distributions.	**5 Hours**
Module -4	

Processing and analysis of Data Statistical measures and their significance: Central tendencies, variation, skewness, Kurtosis, time series analysis, correlation and regression, Testing of Hypotheses: Parametric (t and Chi Square).	**5 Hours**
Module -5	**6 Marks**
Essential of Report writing and Ethical issues: Significance of Report Writing, Different Steps in Writing Report, Layout of the Research Report, Ethical issues related to Research, Plagiarism and self-Plagiarism, Publishing**.**	

Question paper pattern:

- There will be 10 questions with 2 full questions from each module.
- Each full question consists of 20 marks.
- Students have to answer 5 full questions, selecting ONE from each module.

Text Books:

1. Kothari C.R., Research Methodology Methods and techniques by, New Age International Publishers, 3rd Edition, 2013.

Reference Books:

1. Krishnaswami K N, Sivakumar A I and Mathirajan M, "Management Research Methodology", Pearson Education, 2006.
2. 2. Levin R I and Rubin D S, Statistics for Management, 7th Edition, Pearson Education, 2008.

Course Outcomes (CO):

CO1: Understand various research objectives and concepts of qualitative and quantitative research problems and report writing.
CO2: Apply appropriate method for data collection, process the complex data and prepare a report.
CO3: Analyse the real word data with quantitative techniques and interpret the results.
CO4: Formulate research methodology for real world problems.

Cos	**Mapping with POs**
CO1	PO2, PO4, PO8
CO2	PO2, PO3, PO4, PO5
CO3	PO5, PO6
CO4	PO1, PO2, PO3, PO4, PO5, PO7, PO10

ENTERPRISE APPLICATIONS LABORATORY IV SEMESTER			
Laboratory Code	18MCAL48	**CIE Marks**	50
Number of Lecture Hours/Week	02	**SEE Marks**	50
Total Number of Lecture Hours	26	**SEE Hours**	3
CREDITS – 0:0:1			

Course Learning Objective(CLO) :

- Learn the fundamental of connecting to the database
- Demonstrate server side programming using Servlet , JSP, EJB.
- Design and develop web applications using Spring and Hibernate Framework.

Program Statements	
15.	Write a JAVA Program to insert data into Student DATA BASE and retrieve info based on particular queries (For example update, delete, search etc…).
16.	Write a JAVA Servlet Program to implement a dynamic HTML using Servlet (user name and Password should be accepted using HTML and displayed using a Servlet).
17.	Write a JAVA Servlet Program to Auto Web Page Refresh (Consider a webpage which is displaying Date and time or stock market status. For all such type of pages, you would need to refresh your web page regularly; Java Servlet makes this job easy by providing refresh automatically after a given interval).
18.	Write a JAVA Servlet Program to implement and demonstrate get() and Post methods(Using HTTP Servlet Class).
19.	Write a JAVA Servlet Program using cookies to remember user preferences.
20.	Write a JAVA Servlet program to track HttpSession by accepting user name and password using HTML and display the profile page on successful login.
21.	Write a JAVA JSP Program which uses jsp:include and jsp:forward action to display a Webpage.
22.	Write a JAVA JSP Program which uses <jsp:plugin> tag to run a applet
23.	Write a JAVA JSP Program to get student information through a HTML and create a JAVA Bean class, populate Bean and display the same information through another JSP
24.	Write a JSP program to implement all the attributes of page directive tag.
25.	Develop JDBC application using Spring framework
26.	Develop MVC application using Spring framework

Note 1: In the practical Examination each student has to pick one question from a lot of all the 13 questions.

Course outcomes(CO):

CO: Design and Develop real time applications using Servlets & Springs Framework.

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO	PO1,PO2,PO4,PO5,PO8,PO11

SEMESTER –IV **Advanced Web Technologies Laboratory**			
Laboratory Code	18MCAL48	CIE Marks	50
Number of Lecture Hours/Week	02	SEE Marks	50
Total Number of Lecture Hours	26	SEE Hours	03
CREDITS – 0:0:1			

Course Learning Objectives(CLO):

- To understand and analyses the role of server side scripting languages.
- To develop web applications using PHP, Ajax & Angular.
- To build responsive web application using bootstrap components and enhance with jquery effects.

Note : Student has to pick one question during examination.

PART A

1	Design a web page using Collapse, Cards and Badges.
2	Design a webpage with Home tab and Sign In links using Tabs. Apply modal for Sign In page and an image for Home tab.
3	a. Design a web page for Photo Gallery using Bootstrap Carousel b. Design a web page using Date picker and tooltips.
4	a) Write a PHP program to store current date-time in a COOKIE and display the 'Last visited on' date-time on the web page upon reopening of the same page. b) Write a PHP program to store page views count in SESSION, to increment the count on each refresh, and to show the count on web page.
5	Design a web page using Angular Controllers and Directives.
6	Write jQuery program to solve the following : a) Limit character input in the text area including count. b) Based on check box, disable / enables the form submit button.
7	Design a single page web application using Angular & NodeJs.
8	Design an asynchronous web application using Ajax to send data to the server using GET/POST method.
9	Create XHTML form with Name of License Holder, Gender, Vehicle ID , License plate and Date of Model. On submitting, store the values in MySQL table. Retrieve and display the data based on name.
10	Develop a web page using PHP –Ajax that can communicate with a web server when user type characters in an input field (Search Suggest)

Course Outcome (CO):

CO: Design a single page web application and develop asynchronous web application using PHP, Ajax with MYSQL database.

Course Outcomes(CO)	Mapping with Program Outcomes(PO)
CO-1	PO4,PO5,PO11

MOBILE APPLICATIONS USING ANDROID LABORATORY

SEMESTER – IV

Course Code	18MCAL49	**CIE Marks**	50
Number of Practical Hours/Week and Number of Instructional Hours/Week	4	**SEE Marks**	50
Total Number of Lecture Hours	52	**SEE Hours**	03

CREDITS – 0:1:1

Course Learning Objectives:

- Learn the basics of mobile app development
- Build mobile applications using database
- Develop mobile app that uses GPS location information
- Students will learn to develop a mobile app project using multiple features learnt

Laboratory Programs:

The laboratory can be carried out only using any mobile application software.

Note:

1. Students are required to execute one question from Part A and give demo from Part B.
2. Part A has to be evaluated for 20 marks and Part B has to be evaluated for 30 marks along with the report.
3. The project should be carried out with a team strength of maximum two.
4. Students are expected to work for mini project apart from lab hours also with the contact of guides.

PART – A

1. Demonstrate layout
2. Demonstrate widgets
3. Demonstrate life cycle
4. Demonstrate Intents in Android, Shared preferences
5. Demonstrate Fragments in android
6. Demonstrate Animations
7. Demonstrate Email and SMS sending
8. Demonstrate Databases and content providers

   (Database using Silverlight , MySQL and firebase)
9. Demonstrate Services
10. Demonstrate Sensors and location based services
11. Demonstrate Audio playback and image capture

PART – B

Mini-Project

Students should be able to build a complete mobile app using multiple features learnt in Part – A with user interfaces and database connectivity. The Project should be deployed on the cloud like any cloud tool (ex.MS Azure, AWS etc..).

The team must submit a brief project report (25-30 pages) that must include the following

a. Introduction
b. Requirement Analysis
c. Software Requirement Specification
d. Analysis and Design
e. Implementation
f. Testing

4. The report must be evaluated for 10 Marks. Demonstration and Viva for 20 Marks.

Course Outcome:

CO1: Design and develop various android mobile applications.

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO	PO1,PO2,PO4,PO5,PO8,PO11

Dr. Ambedkar Institute of Technology, Bengaluru-56

(An Autonomous Institute, Affiliated to VTU, Belagavi)

Master of Computer Applications Program

(Accredited by National Board of Accreditation)

MCA V – VI semester Syllabus
2018 Scheme

Dr. Ambedkar Institute of Technology

(An Autonomous Institute affiliated to VTU, Accredited by NAAC with 'A' grade)

Department of Master of Computer Applications

SCHEME OF TEACHING AND EXAMINATION OF MCA FIFTH SEMESTER (AUTONOMOUS) 2018-2021

		V SEMESTER							
Subject Code	Name of the Subject	Credits			Total Credits	Exam Duration in Hours	Marks		Total Marks
		Lecture	Tutorial	Practicals			CIE	SEE	
18MCA51	Machine Learning using Python	4	-	-	4	3	50	50	100
18MCA52	Big Data Analytics	3	-	-	3	3	50	50	100
18MCA53	Cloud Computing	3	-	-	3	3	50	50	100
18MCA54	Elective -4	3	-	-	3	3	50	50	100
18MCA55	Elective -5	3	-	-	3	3	50	50	100
18MCAL56	Machine Learning using Python Lab	--	-	1	1	3	50	50	100
18MCAL57	Big Data Analytics Lab	--	-	1	1	3	50	50	100
18MCAL58	Cloud Computing Lab	--	-	1	1	3	50	50	100
18MCAI59	Industry Internship (6 weeks)	--	--	--	5	3	50	50	100
	Total	16	-	3	24	27	450	450	900

ELECTIVE-4									
V SEMESTER									
Subject Code	**Name of the Subject**	**Credits**				**Exam Duration**	**Marks**		**Total Marks**
		L	**T**	**P**	**Credits**		**CIE**	**SEE**	
18MCA541	**Enterprise Application-2**	**3**	**-**	**--**	**3**	**3**	**50**	**50**	**100**
18MCA542	**Full Stack Development with MERN**	**3**	**-**	**--**	**3**	**3**	**50**	**50**	**100**
18MCA543	**DevOps**	**3**	**-**	**--**	**3**	**3**	**50**	**50**	**100**

Dr. Ambedkar Institute of Technology

(An Autonomous Institute affiliated to VTU, Accredited by NAAC with 'A' grade)

Department of Master of Computer Applications

SCHEME OF TEACHING AND EXAMINATION OF MCA SIXTH SEMESTER
(AUTONOMOUS) 2018-2021

VI SEMESTER							
Subject Code	**Name of the Subject**	**Internal**	**External**	**Dissertation Evaluation**	**Viva voce**	**Total Marks**	**Credits**
18MCAS61	**Technical Seminar**	**50**	**--**	**--**	**--**	**50**	**2**
18MCAP62	**Project Work**	**50**	**-**	**100**	**100**	**250**	**20**
Total		**100**	**--**	**100**	**100**	**300**	**22**

V SEMESTER

MACHINE LEARNING USING PYTHON

Course code:				**18MCA51**		**CIE Marks:**	**50**
Number of Lecture Hours per week:				**4**		**SEE Marks:**	**50**
Total number of Lecture Hours:				**52**		**SEE Hours:**	**3**
Lecture (L):	**4**	**Practical (P):**		**Tutorial (T):**		**Total Credits:**	**4**

COURSE LEARNING OBJECTIVES (CLO)

- **To distinguish between, supervised & unsupervised and Gain knowledge about basic concepts of Machine Learning.**
- **To introduce participants to the fundamentals of data analytics using Python**
- **To apply the appropriate machine learning strategy for any given problem.**
- **To develop skills of using recent machine learning software for solving practical problems.**

MODULES	TEACHING HOURS
MODULE 1: Introduction to Machine learning	**10 Hrs**
Introduction toMachine Learning, types of Machine learning, Applications, Machine Learning Process, Well posed learning problems, Designing a Learning system, Perspective and Issues in Machine Learning	
MODULE 2: Python for Machine Learning	**11 Hrs**
Introduction to Pandas Data structures, Function Application & Mapping, Correlation & Covariance, Handling Missing Data, Reading & Writing Data in CSV or text files, Data Preparation-Merging and Removing data, Data Transformation-Removing Duplicates, Mapping.	
MODULE 3: Concept Learning & Unsupervised Learning	**10 Hrs**
Introduction to Bayes Theorem and Concept learning, Naive Bayes Classifier, Applications of Naïve Bayes Classifier, Clustering –Different types of the clustering techniques, K-Means Clustering	
MODULE 4: Supervised Learning	**11 Hrs**
Training a model-Linear Regression, Multiple Linear regression, Improving accuracy of Linear Regression Model, Polynomial Regression ModelClassification-Introduction, Decision Tree, Random Forest Model, Support Vector Machines, Boosting	
MODULE 5: Neural Network and Deep Learning	**10 Hrs**

Artificial Neural Networks: IntroductionArtificial Neural Networks: Introduction, Neural Network representation, Appropriate problems, Perceptrons, Back propagation algorithm. Deep Learning-Introduction, Deep Learning Architectures	

Question Paper Pattern:

- **Each full question consists of 20 marks.**
- **Questions are set covering all the topics under each module**

TextBooks

1. Fabio Nelli, " Python Data Analytics", Apress,Springer Science + Business Media Finance Inc (SSBM Finance Inc).
2. Machine Learning, SaikatDutt, Subramanian Chandramouli, Amit Kumar Das, 1st Edition, 2019, Pearson Publications, , ISBN 978-93-530-6669-7
3. Machine Learning, Tom M Mitchel, McGraw Hill publications,ISBN-0070428077
4. Machine Learning with Python: Design and Develop Machine Learning and Deep Learning, BPB Publishing, India,2018

Reference Books

1. Jake Vander plas, "Python Data Science Handbook: Essential tools for working with data", O'Reilly Publishers, I Edition.
2. EthemAlpaydin "Introduction To Machine Learning" 2nd Edition PHI Learning Pvt. Ltd-New Delhi.

COURSE OUTCOMES (CO)

CO1: Understand the concepts related to Machine Learning techniques.

CO2: Demonstrate Pre-processing techniques and perform exploratory data analysis related to a scenario.

CO3:Identify and apply the appropriate techniques to process the data and solve the applications using machine learning techniques

CO4: Apply data analytics principles and techniques of Machine learning to solve real time problems

Course Outcomes(COs)	**Mapping with Program Outcomes(POs)**
CO1	**PO1,PO2,**
CO2	**PO1,PO2,PO4,PO8**
CO3	**PO1,PO2,PO4,PO5,PO8,PO10,PO11**
CO4	**PO1,PO2,PO4,PO5,PO8,PO10,PO11**

V SEMESTER

BIG DATA ANALYTICS

Course code:				**18MCA52**		**CIE Marks:**	**50**
Number of Lecture Hours per week:				**3**		**SEE Marks:**	**50**
Total number of Lecture Hours:				**39**		**SEE Hours:**	**3**
Lecture (L):	**3**	**Practical (P):**		**Tutorial (T):**		**Total Credits:**	**3**

COURSE LEARNING OBJECTIVES (CLO)

- **To impart fundamental concepts about big data and its identification.**
- **To analyse the design of Hadoop Distributed Files system.**
- **To understand and analyse Map Reduce technique for solving Big Data problems**
- **To analyse different hadoop related tools like Pig & Hive**

MODULES	**TEACHING HOURS**
MODULE 1: Big Data & Hadoop Eco system	**8 Hrs**
Example Applications, Basic Nomenclature, Analysis Process Model, Analytical Model Requirements , types of Data Sources, Sampling, Types of data elements, data explorations, exploratory statistical analysis, missing values, outlier detection and Treatment, cloud and Big Data –Predictive Analytics. A Brief History of Hadoop, Apache Hadoop and the Hadoop Ecosystem Hadoop Releases Response.	
MODULE 2: The Hadoop Distributed File system	**8 Hrs**
The Hadoop Distributed File system The Design of HDFS, HDFS Concepts, Blocks, Name nodes and Datanodes, HDFS Federation, HDFS High-Availability, The Command Line Interface, Basic File system Operations, Hadoop File systems Interfaces ,The Java Interface, Reading Data from a Hadoop URL, Reading Data Using the File System API, Writing Data, Directories, Querying the File system, Deleting Data, Data Flow Anatomy of a File Read ,Anatomy of a File Write	
MODULE 3: Map Reduce	**8 Hrs**
A Weather Dataset ,Data Format, Analyzing the Data with Unix Tools, Analyzing the Data with Hadoop, Map and Reduce, Working of Map Reduce - Anatomy of a Map Reduce Job Run, Failures, Shuffle and Sort, Task Execution, Map Reduce Formats - Input Formats, Output Formats	
MODULE 4: Hadoop Tool-Pig	**8 Hrs**
Pig – Grunt – pig data model – Pig Latin – developing and testing Pig Latin scripts, **Pig Latin** – Structure, Statements, Expressions, Types, Schemas, Functions, Macros,User-Defined Functions – A Filter UDF, An Eval UDF, A Load UDF.DataProcessing Operators – Loading and storing of data, Filtering data,	

Groupingand Joining data, Sorting data	
MODULE 5: Hadoop Tool-Hive	**7 Hrs**
Installing Hive – The Hive shell,Hive – Architecture, data types and file formats – HiveQL data definition – HiveQL data manipulation – HiveQL queries. Tables – Managed Tables and External Tables, Partitions and Buckets, Importing Data,Querying Data – Sorting and Aggregating,Storage Formats, Joins, Sub queries, Views.	

Question Paper Pattern:

- **Each full question consists of 20 marks.**
- **Questions are set covering all the topics under each module**

TextBooks

1. Bart Baesens, " Analytics in a Big Data World : The Essential Guide to Data Science and its Applications" Wiley
2. Tom White, "Hadoop: The Definitive Guide", 3rd Edition, O'reilly, 2012.
3. E. Capriolo, D. Wampler, and J. Rutherglen, "Programming Hive", O'Reilley, 2012.
4. Alan Gates, "Programming Pig", O'Reilley, 2011

Reference Books

1. Boris lublinsky, Kevin t. Smith, Alexey Yakubovich, "Professional Hadoop Solutions", Wiley, ISBN: 9788126551071, 2015.
2. Vignesh Prajapati, Big data analytics with R and Hadoop, SPD 2013.

COURSE OUTCOMES (CO)

CO1: Explain the fundamentals of big data analytical techniques and usage of hadoop tools.

CO2: Analyse Hadoop ecosystem and Map Reduce concept to solve big data problems.

CO3: Design a Map-Reduce model to process the datausing tools for a use case.

CO4: Evaluate the performance of data analytics and visualize the results.

Course Outcomes(COs)	**Mapping with Program Outcomes(POs)**
CO1	**PO1,PO4**
CO2	**PO2,PO4,PO5**
CO3	**PO2,PO3,PO4,PO5,PO7,PO11**
CO4	**PO2,PO3,PO4,PO5,PO7,PO11**

V SEMESTER CLOUD COMPUTING							
Course code:				**18MCA53**		**CIE Marks:**	**50**
Number of Lecture Hours per week:				**03**		**SEE Marks:**	**50**
Total number of Lecture Hours:				**39**		**SEE Hours:**	**3**
Lecture (L):	**3**	**Practical (P):**	**0**	**Tutorial (T):**	**0**	**Total Credits:**	**3**

COURSE LEARNING OBJECTIVES (CLO)

- **Introduce the fundamental aspects of cloud computing**
- **Discuss virtualization technologies along with the architectural models of cloud computing.**
- **Leverage the prominent Cloud computing technologies available in the market place.**
- **Demonstrate different features of cloud platforms used in Industry**
- **To understand how energy efficiency achieved in cloud computing using green computing and understand the mechanism needed to harness cloud computing in the respective endeavours**

MODULES	**TEACHING HOURS**
MODULE-1: CLOUD COMPUTING OVERVIEW	**07 Hours**
Cloud Computing Overview, The Vision of Cloud Computing, Defining a Cloud, A Closer Look, Cloud Computing Architecture, Characteristics and Benefits, Challenges in the cloud, Historical Developments, Distributed Systems, Virtualization, Web 20, Service Oriented Computing, Utility-Oriented Computing, Building Cloud Computing Environments, Application Development, Infrastructure and System Development, Computing Platforms and Technologies, Amazon Web Services (AWS), Google AppEngine, Microsoft Azure, Hadoop, Forcecom and Salesforcecom,	
Module-2: Virtualization	**8 Hours**

Virtualization Introduction, Characteristics of virtualized environments, Increased security, Managed execution, Portability, Taxonomy of virtualization techniques, Virtualization and cloud computing, Pros and cons of virtualization, Technology examples- Xen par virtualization, VMware: full virtualization, Microsoft Hyper-V. Cloud Computing Architecture: Introduction, Reference model- Architecture, Infrastructure- and hardware-as-a-service, Platform as a service, Software asa service, Deployment Model- Public clouds, Private clouds, Hybrid clouds, Community clouds, Open challenges.	
Module-3:CloudManagement	**9 Hours**
Service Level Agreement, Cloud Economics, Managing Data, Introduction to Map Reduce, Open Stack, Resource Management.	
Module-4: Cloud Platforms in Industry	**08 Hours**
Amazon web services: Compute services, Storage services, Communication services, Additional services. Google Cloud, AppEngine: Architecture and core concepts, Application life cycle, Cost model Observations Microsoft Azure: Azure core concepts, SQL Azure, Windows Azure platform appliance, Observations	
Module-5: Advanced Topics in Cloud Computing	**08Hours**
Green cloud computing , Introduction to Docker Container, Sensor Cloud Computing, IoT Cloud, Fog Computing, Mobile Cloud Computing	
Question Paper Pattern: • **Each full question consists of 20 marks.** • **Questions are set covering all the topics under each module**	
Textbooks:	
1. Cloud Computing: Principles and Paradigms, Editors: Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Wiley,2011	
2. Enterprise Cloud Computing - Technology, Architecture, Applications, Gautam Shroff, Cambridge University Press, 2010	
3. Cloud Computing Bible, Barrie Sosinsky, Wiley-India, 2010	
Reference Books	
1. Buyya, Rajkumar, James Broberg, and Andrzej M. Goscinski, eds. Cloud computing: Principles and paradigms. Vol. 87. John Wiley & Sons, 2010.	
COURSE OUTCOMES (CO)	
CO1: Explain the fundamental principles of cloud computing and its related Concepts.	
CO2: Analyze Prominent Cloud computing technologies available in the marketplace.	
CO3: Apply suitable applications to leverage the strength of cloud computing.	

CO4: Develop the applications of cloud Computing that can harness the power of cloud computing.	
CO5: Explain the fundamental principles of cloud computing and its related Concepts.	
COURSE OUTCOMES MAPPING WITH PROGRAM OUTCOMES:	
Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO1	**PO1,PO2,PO3,PO4**
CO2	**PO1,PO2,PO3,PO4, PO6**
CO3	**PO1,PO2,PO3,PO4,PO6**
CO4	**PO1,PO2,PO3,PO4**
CO1	**PO1,PO2,PO3,PO4**

V SEMESTER

MACHINE LEARNING USING PYTHON LAB

Course code:				**18MCAL56**		**CIE Marks:**	**50**
Number of Lecture Hours per week:				**2**		**SEE Marks:**	**50**
Total number of Lecture Hours:				**26**		**SEE Hours:**	**3**
Lecture (L):		**Practical (P):**	**2**	**Tutorial (T):**	**0**	**Total Credits:**	**1**

COURSE LEARNING OBJECTIVES (CLO)

- To understand Pre-processing techniques and perform exploratory data analysis .
- Identify and apply Machine Learning algorithms to solve real world problems
- To develop skills of using recent machine learning software for solving practical problems

Sl.No	Program
1.	Create a Data frame and demonstrate different ways to treat missing values.
2.	Implement Data Wrangling (Merge, Concatenate, Group) and Data Aggregation.
3.	a. Write a python program to read and write data into files (.CSV, .txt, .XLS). b. Perform exploratory data analysis (Head, Tail, Description, etc.) on any dataset.
4.	Implement Linear Regression using Python Script and identify explanatory variables.
5.	Write a program to demonstrate the working of the decision tree.
6.	Implement clustering technique for a given data set in python.
7.	Write a program to implement the naïve Bayesian classifier for a sample training data set stored asa .CSV file. Compute the accuracy of the classifier, considering few test data sets.
8.	Build an Artificial Neural Network by implementing the Back propagation algorithm and test thesame using appropriate data sets.

Note : Student has to pick one question from a lot of 8 questions

COURSE OUTCOMES (CO)

CO1: Implement exploratory data analysis, data visualization and different machine Learning Techniques to solve real world problems in Python.

Course Outcomes(COs)	Mapping with Program Outcomes(POs)
CO1	**PO2, PO4, PO5, PO7,PO11**

V SEMESTER
BIG DATA ANALYTICS LAB

Course code:				**18MCAL57**		**CIE Marks:**	**50**
Number of Lecture Hours per week:				**2**		**SEE Marks:**	**50**
Total number of Lecture Hours:				**26**		**SEE Hours:**	**3**
Lecture (L):	**0**	**Practical (P):**	**2 Hrs**	**Tutorial (T):**	**0**	**Total Credits:**	**1**

COURSE LEARNING OBJECTIVES (CLO)

- To set up single and multi-node Hadoop Clusters.
- To solve Big Data problems using Map Reduce Technique.
- To design algorithms that uses Map Reduce Technique to apply on Unstructured and structured data.
- To implement programming tools PIG and HIVE in Hadoop eco system.

Sl.No	Program
1.	Hadoop Installation.
2.	Installation of VMWare to setup the Hadoop environment and its ecosystems.
3.	a. Implement the following file management tasks in Hadoop: i. Adding files and directories ii. Retrieving files iii. Deleting files
4.	Run a basic word count Map Reduce program to understand Map Reduce Paradigm.
5.	Write a Map Reduce program that mines weather data.
6.	Implement matrix multiplication with Hadoop Map Reduce.
7.	Installation of PIG. Write Pig Latin scripts sort, group, join, project, and filter your data.
8.	a. Run the Pig Latin Scripts to find Word Count b. Run the Pig Latin Scripts to find a max temp for each and every year.
9.	Installation of HIVE. Use Hive to create, alter, and drop databases, tables, views, functions, and indexes.

Note : Student has to pick one question from a lot of 9 questions

COURSE OUTCOMES (CO)

CO1: Apply Hadoop, MapReduce, HDFS and YARN develop big data applications and Explore the working of Pig & Hive and analyse the results.

Course Outcomes(COs)	Mapping with Program Outcomes(POs)
CO1	**PO3,PO4,PO5,PO7,PO11**

V SEMESTER
CLOUD COMPUTING LAB

Course code:				**18MCAL58**		**CIE Marks:**	**50**
Number of Lecture Hours per week:				**2**		**SEE Marks:**	**50**
Total number of Lecture Hours:				**26**		**SEE Hours:**	**3**
Lecture (L):		**Practicals (P):**	**2 Hrs**	**Tutorial (T):**	**0**	**Total Credits:**	**1**

COURSE LEARNING OBJECTIVES (CLO)

- To work with Virtualization.
- Explore different Cloud services: Amazon, Google apps and Salesforce and VMware
- Design Virtual Machine using VM player and test Client server application using Virtual Machine.
- Demonstrating IaaS, PaaS and SaaS.

Sl. No	Program
10.	• **Working with Amazon Web Services(AWS)** • Familiarize the services by AWS • Creating user login • Creating Linux, Windows virtual machines instance using EC2 • Run simple applications on EC2 Instance • Creating Storage using S3 • Create a Backup using Image and launch new instance using Backup image • Creating an RDS Instance with MySQL Workbench and Dynamo DB • Demonstrate Database application on AWS • Upgrading and downgrading the infrastructure based on the requirement • Demonstrate Load balancing using different instance of EC2 • Launch a web application. • Demonstration of Identity and Access management.

	• Demonstrate Elastic bean stack • Demonstrate AWS dynamic web application
11.	**SalesforceTrailhead Platform** • Create a web application to enter the students' details like name, USN, semester, section and CGPA to a database on Salesforce cloud platform. • Create a web application to implement an online cart for adding items to a shopping cart and deleting it. • Create a web application to enter the faculty details like faculty ID, faculty name, and salary to a database and calculate the income tax to be paid by the faculty at the end of financial year. • Create a web application to book a flight from a source to destination and store the status of flight, and departure timings on database. • Create a Collaborative learning environment for a particular learning topic using Google Apps. Google Drive, Google Docs and Google Slides must be used for hosting e-books, important articles and presentations respectively. • Develop Department events' registration app with an object containing event name, date/time, venue as parent relationship, another object containing student name, branch, event name, date/time, and venue as child relationship. • Develop Blood donation registration app with an object which records donors' name , age and blood group as parent relationship and another object containing hemoglobin level, donated or not details (if age>18) child relationship. • Develop Attendance maintenance app with an object to record student details, attendance and provide a link to college websites' results webpage. • Create a web application with objects to maintain database of an art gallery which contains objects like artists, arts, and inventory and provide a link to any of the art gallery website.

Course Outcome(CO)

CO1: Demonstrate Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS).

COURSE OUTCOMES	**MAPPING WITH PROGRAM OUTCOMES:**
CO 1	**PO1,PO2,PO3**

V SEMESTER
ENTERPRISE APPLICATION-2

Course code:				**18MCA541**		**CIE Marks:**	**50**
Number of Lecture Hours per week:				**3**		**SEE Marks:**	**50**
Total number of Lecture Hours:				**39**		**SEE Hours:**	**3 Hrs**
Lecture (L):	**3**	**Practicals (P):**	**0**	**Tutorial (T):**	**0**	**Total Credits:**	**3**

COURSE LEARNING OBJECTIVES (CLO)

- **To describe the Fundamentals of .Net framework**
- **To demonstrate Object Oriented Programming concepts using C#**
- **To implement delegates, event handling and exception handling**
- **To develop Web applications using ASP.NET,ADO.NET**

MODULES	TEACHING HOURS
MODULE 1: Getting started with .NET Framework 4.0 and C# Understanding Previous Technologies, Benefits of .NET Framework, Architecture of .NET Framework 4.0,.NET Execution Engine, Components of .NET Framework 4.0: CLR, CTS, Metadata and Assemblies, .NET Framework Class Library, Windows Forms, ASP .NET and ASP .NET AJAX, ADO .NET, Windows workflow Foundation, Windows Presentation Foundation, Windows Communication Foundation, Widows Card Space and LINQ. **Introducing C#** Need of C#, C# Pre-processor Directives, Creating a Simple C# Console Application, Identifiers and Keywords. Data Types, Variables and Constants: Value Types, Reference Types, Type Conversions, Boxing andUnBoxing , Variables and Constants . Expression and Operators : Operator Precedence, Using the ?? (Null Coalescing) Operator, Using the :: (Scope Resolution) Operator and Using the is and as Operators. Control Flow statements: Selection Statements, Iteration Statements and Jump Statements	**7 Hrs**
MODULE 2: Namespaces,Classes and Object Oriented Programming Namespaces, The System namespace, Classes and Objects: Creating a Class, Creating an Object, Using this Keyword, Creating an Array of Objects, Using the Nested Classes, Defining Partial Classes and Method, Returning a Value from a Method and Describing Access Modifiers. Static Classes and Static Class Members. Properties: Read-only Property, Static Property, Accessibility of assessors and Anonymous types. Indexers, Structs: Syntax of a struct and Access Modifiers for structs.System.Object Class **Encapsulation:** Encapsulation using assessors and mutators, Encapsulation using Properties. Inheritance: Inheritance and Constructors, Sealed Classes and Sealed Methods, Extension methods. **Polymorphism:** Compile time Polymorphism/ Overloading, Runtime Polymorphism/ Overriding. Abstraction: Abstract classes, Abstract methods.	**8 Hrs**

Interfaces: Syntax of Interfaces, Implementation of Interfaces and Inheritance	
MODULE 3: Delegates, Events, Exception Handling **Delegates:**Creating and using Delegates, Multicasting with Delegates. **Events:** Event Sources, Event Handlers, Events and Delegates, Multiple Event Handlers. **Exception Handling**: The try/catch/throw/finally statement, Custom Exception. System. Exception, Handling Multiple Exception	**8 Hrs**
MODULE 4: Graphical User Interface with Windows Forms Introduction, Windows Forms, Event Handling: A Simple Event- Driven GUI, Control Properties and Layout, Labels, TextBoxes and Buttons, GroupBoxes and Panels, CheckBoxes and RadioButtons, ToolTips, Mouse-Event Handling, Keyboard-Event Handling. Menus, Month Calendar Control, LinkLabel Control, ListBox Control, ComboBox Control, TreeView Control, ListView Control, TabControl and Multiple Document Interface (MDI) Windows.	**8 Hrs**
MODULE 5: Web App Development and Data Access using ADO.NET Introduction to Web Basics, Multitier Application Architecture, First Web Application: Building Web-Time Application, Examining Web-Time.aspx's Code-Behind File, Understanding Master pages, Standard Web Controls: Designing a Form, Validation Controls, GridView Control, DropDownList, Session Tracking.Set up the sample database,Create the forms and add controls,Store the connection string,Retrieve the connection string,Write the code for the forms,Test your application **ASP.NET AJAX :**Exploring AJAX,Need for AJAX, AJAX and other Technologies, AJAX Server Controls, ScriptManager control, Update Panel, UpdateProgress Control, Creating Simple Application using AJAX Server Controls.	**8 Hrs**
Question Paper Pattern: • **Each full question consists of 20 marks.** • **Questions are set covering all the topics under each module**	
TextBooks:	
1. .NET 4.0 Programming (6-in-1), Black Book, Kogent Learning Solutions Inc., Wiley-Dream Tech Press.	
2. Paul Deitel and Harvey Deitel: C# 2010 for Programmers, 4th Edition, Pearson Education.	
Reference Books	
1. Andrew Trolsen: Pro C# 5.0 and the .NET 4.5 Framework, 6th Edition, WileyAppress.	
2. Bart De Smet: C# 4.0 Unleashed, Pearson Education- SAMS Series.	
3. Herbert Schildt: Complete Reference C# 4.0, Tata McGraw Hill, 2010.	
COURSE OUTCOMES (CO)	

CO1: Distinguish the features of C# and client-server concepts using .Net Framework Components.	
CO2: Demonstrate delegates, events and exception handling with ASP, Win Form, ADO.NET.	
CO3: Develop Graphical User Interface for various applications	
CO4: Develop Web based and Console based applications with database connectivity	
COURSE OUTCOMES MAPPING WITH PROGRAM OUTCOMES:	
Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO1	**PO1,PO2,PO3,PO4,PO5,PO8**
CO2	**PO1,PO2,PO3,PO4**
CO3	**PO1,PO2,PO3,PO4,PO5**
CO4	**PO1,PO2,PO3,PO4,PO5**

V SEMESTER

FULL STACK DEVELOPMENT WITH MERN

Course Code:				**18MCA542**		**CIE Marks:**	**50**
Number of Lecture Hours per week:				**3**		**SEE Marks:**	**50**
Total number of Lecture Hours:				**39**		**SEE Hours:**	**3**
Lecture (L):	**3**	**Practicals (P):**	**0**	**Tutorial (T):**	**0**	**Total Credits:**	**3**

COURSE LEARNING OBJECTIVES (CLO)

- **To design as web page using front end technologies**
- **To develop application with server side scripting tools**
- **To develop web application with REST APIs and use of framework to communicate client-server applications.**
- **To build as responsive web application with managing NOSQL databases.**

MODULES	TEACHING HOURS
MODULE 1: Introduction to React	**10 Hrs**
Welcome to React: Obstacles and Roadblocks, React's future, keeping up with the changes, working with the files. The Basics-Introduction, Installation, getting started hello world program, Lifecycle of Components, Understanding Functional & Class Components Passing Data.	
MODULE 2: React Components and Redux	**11 Hrs**
React Props, React state-setting state, Event handling, Designing components-state vs props An Introduction to Redux- Core Concepts, Reducer, Action, Action Creator, Combining Reducers, Store, Data Flow in Redux, Usage with React	
MODULE 3: Programming in Node.js	**11 Hrs**
Node.js Installation –getting started, Control flow, asynchronous pattern callback, Sequential functionality, nested callbacks and exception handling, asynchronous patterns and control flow. Routing Traffic, Serving Files and Middleware: Building a Simple Static File Server from Scratch, Middleware, Routers and Proxies	
MODULE 4: Expressing REST APIs	**10 Hrs**
REST-HTTP Methods as actions, Express-Routing, Handler Functions, The List API-automatic Server Restart, testing, Create API, Error Handling.	
MODULE 5: Module Title	**10 Hrs**

Introduction to MongoDB: -Installation-Databases, Data Types, Using MongoDB Shell. Creating, Updating, Deleting and Querying Documents: Inserting, removing, and updating the documents. Scheme Initialization, Reading and writing to Mongodb.	

Question Paper Pattern:

- **Each full question consists of 20 marks.**
- **Questions are set covering all the topics under each module**

Text Books

1. Tomasz Dyl Kamil Przeorski, "Mastering Full-Stack React Web Development", 2017 Packt Publishing
2. Vasan Subramanian ,"ProMERN Stack",Apress,2018.

Reference Books

1. Eddy Wilson IriarteKoroliova ,"MERN-Full stack Development", Packt Publishing Ltd.,2018
2. ShamaHoque,"Full stack React Projects",Pack Publishing Ltd.,2018.

COURSE OUTCOMES (CO)

CO1: Demonstrate basic concepts of react, node, express and mongodbtechologies

CO2: Design front end application using React and Redux libraries.

CO3: Develop interactive web applications on server side with NOSQL databases.

CO4: Build responsive web application communicating with RES API and managing data with NOSQL databases.

Course Outcomes(Cos)	Mapping with Program Outcomes(POs)
CO1	**PO5,PO11**
CO2	**PO2,PO4,PO5,PO11**
CO3	**PO2,PO4,PO5,PO7,PO11**
CO4	**PO5,PO11**

V SEMESTER DEVOPS							
Subject Code			**18MCA543**		**CIE Marks**		**50**
Number of Lecture Hours/Week			**03**		**SEE Marks**		**50**
Total Number of Lecture Hours			**39**		**SEE Hours**		**03**
Lecture (L):	**03**	**Practical (P):**		**Tutorial (T):**	**-**	**Total Credits:**	**3**

Course Learning objectives(CLO):

- Introduce devopps concepts and architecture of Devopps
- Analyze Building the code and deployment

MODULES	TEACHING HOURS
Module -1 Introduction Introduction to DevOps and Continuous Delivery ,Introducing DevOps, How fast is fast?,The Agile wheel of wheelsBeware the cargo cult Agile fallacy, DevOps and ITIL.The DevOps process and Continuous Delivery – an overview,The developers, The revision control system, The build server, The artifact repository,Package managers, Test environments, Staging/production, Release management, Scrum, Kanban, and the delivery pipeline, Wrapping up – a complete example, Identifying bottlenecks.	**7 Hours**
Module -2 DevOpps Architecture How DevOps Affects Architecture, Introducing software architecture, The monolithic scenario, Architecture rules of thumb, The separation of concerns, The principle of cohesion, Coupling, Back to the monolithic scenario, A practical example, Three-tier systems, The presentation tier, The logic tier, The data tier, Handling database migrations, Rolling upgrades, Hello world in Liquibase, The changelog file, The pom.xml file, Manual installation, Microservices, Interlude – Conway's Law, How to keep service interfaces forward compatible, Microservices and the data tier, DevOps, architecture, and resilience.	**8 Hours**
Module – 3 Building the code Why do we build code?, The many faces of build systems, The Jenkins build server, Managing build dependencies, The final artifact, Cheating with FPM, Continuous Integration, Continuous Delivery, Jenkins plugins, The host server, Build slaves, Software on the host, Triggers, Job chaining and build pipelines, A look at the Jenkins filesystem layout, Build servers and infrastructure as code, Building by dependency order, Build phases, Alternative build servers, Collating quality measures, About build status visualization, Taking build errors seriously, Robustness.	**7 Hours**

Module -4Deploying the Code Why are there so many deployment systems?Configuring the base OS,Describing clusters, Delivering packages to a ,system,Virtualizationstacks,Executing code on the client ,A note about the exercises, The Puppet master and Puppet agents, Ansible, PalletOps, Deploying with Chef,Deploying with SaltStack,Salt versus Ansible versus Puppet versus PalletOps execution models,Vagrant,Deploying with Docker,Comparisontables,Cloudsolutions,AWS,Azure	**8 Hours**
Module – 5 Issue Tracking What are issue trackers used for?, Some examples of workflows and issues, What do we need from an issue tracker?, Problems with issue tracker proliferation, All the trackers, Bugzilla, Trac, Redmine, The GitLab issue tracker, Jira. Introducing the IoT and DevOps, The future of the IoT according to the market, Machine-to-machine communication, IoT deployment affects software architecture, IoT deployment security, Okay, but what about DevOps and the IoT again?, A hands-on lab with an IoT device for DevOps.	**8 Hours**

Question Paper Pattern:

- **Each full question consists of 20 marks.**
- **Questions are set covering all the topics under each module**

Text Books:

1. The DevOps Hand Book, Gene Kim, Jez Humble, PatricDebois& John Wills

Reference Books:

1. The Practical Guide to Enterprise DevOps and Continuous Delivery, Julian Fish

Course Outcomes(CO):

CO1: Understand Devops.

CO2: Analyze Architecture .

CO3: Apply how to build the code.

CO4: Deploy the code.

COURSE OUTCOMES MAPPING WITH PROGRAM OUTCOMES

Course Outcomes(CO)	Mapping with Program Outcomes(PO)
CO1	**PO1, PO3**
CO2	**PO1, PO3, PO8**
CO3	**PO1, PO5, PO8**
CO4	**PO1, PO3, PO8, PO10**

SEMESTER-V

WEB SERVICES

Course code				**18MCA551**		**CIE Marks**	**50**
Number of Lecture Hours per week				**4**		**SEE Marks**	**50**
Total number of Lecture Hours				**39**		**SEE Hours**	**3**
Lecture (L):	**4**	**Practicals (P):**	**0**	**Tutorial (T):**	**0**	**Total Credits**	**3**

COURSE LEARNING OBJECTIVES (CLO)

- **To learn the Principles of Web & Web Services**
- **To learn XML Document in both DTD and XML Schema Language.**
- **To design Web Services and its Infrastructure, Use of SOAP, Building a Web Service, Deploying and Publishing Web Service**
- **To Design Web Services Architecture, UDDI Registry**

MODULES	**TEACHING HOURS**
MODULE 1: Introduction to Web Services	**08 Hrs**
The definition of web services, basic operational model of web services benefits and challenges of using web services,Core distributed computing technologies – client/server, CORBA, JAVA RMI, Micro Soft DCOM, MOM, Challenges in distributed computing role of J2EE and XML in distributed computing, emergence of Web Services and Service Oriented Architecture (SOA)	
MODULE 2: Xml Technology	**07 Hrs**
Role Of XML, XML - Name Spaces - Structuring With Schemas and DTD, Presentation Techniques - Transformation - XML Infrastructure	
MODULE 3: SOAP	**08 Hrs**
Core fundamentals of SOAP – SOAP Messages Structure, SOAP encoding SOAP message exchange models SOAP communication and messaging SOAP security, Developing Web Services using SOAP- Building SOAP Web Services, developing SOAP Web Services using Java, limitations Of SOAP	
MODULE 4: Web Services	**08 Hrs**
Overview - Architecture - Key Technologies - WSDL, ebXML - SOAP And Web Services In E-Com - Overview Of .NET And J2EE. – WSDL in the world of Web Services, Web Services life cycle, anatomy of WSDL definition document, WSDL bindings	
MODULE 5: Discovering Web Services	**08 Hrs**
Service discovery, role of service discovery in a SOA service discovery	

mechanisms, UDDI-UDDI Registries, uses of UDDI Registry, Programming with UDDI, UDDI data structures, support for categorization in UDDI registries publishing API Publishing information to a UDDI Registry searching information in UDDI Registry deleting information in a UDDI	

Question Paper Pattern:

- Each full question consists of 20 marks.
- Questions are set covering all the topics under each module

TextBooks:

1. XML, Web Services and the Data Revolution, Frank. P. Coyle, 2012, Pearson Education.
2. Developing Java Web Services, Ramesh Nagappan, Robert Skoczylas and Rima Patel Sriganesh, 2004, Wiley Publishing Inc..

Reference Books

1. Developing Enterprise Web Services, Sandeep Chatterjee, James Webber, 2004, Pearson Education. 2. Java Web Services Architecture, McGovern, et al., 2005, Morgan Kaufmann Publishers

COURSE OUTCOMES (CO)

CO1: To Design Web & Web Services

CO2: To Design the schema for the given XML documents in both DTD and XML Schema languages

CO3: To Building a Web Service, SOAP, Deploying and Publishing Web Service

CO4: Analyze Web Services Architecture, UDDI Registry.

COURSE OUTCOMES MAPPING WITH PROGRAM OUTCOMES:

Course Outcomes(COs)	**Mapping with Program Outcomes(POs)**
CO1	**PO1, PO2, PO5,P10**
CO2	**PO1, PO2**
CO3	**PO1, PO2, PO4, PO5**
CO4	**PO1, PO2, PO5**

V SEMESTER

INTERNET OF THINGS

Course code:				**18MCA552**		**CIE Marks:**	**50**
Number of Lecture Hours per week:				**03**		**SEE Marks:**	**50**
Total number of Lecture Hours:				**39**		**SEE Hours:**	**3 Hrs**
Lecture (L):	**3**	**Practicals (P):**	**0**	**Tutorial (T):**	**0**	**Total Credits:**	**3**

COURSE LEARNING OBJECTIVES (CLO)

- **Learn the evolution of IOT from M2M to global Context**
- **Understand IoT in managing data and knowledge**
- **Analyze the architecture of IoT and assess its industrial applications**

MODULES	TEACHING HOURS
MODULE 1: Understanding M2M and IoT Introduction: The Vision- From M2M to IoT, M2M towards IoT-the global context, A use case example, Differing Characteristics	**8 Hrs**
MODULE 2: Introduction to IoT Architecture **A Market Perspective** – Introduction, Some Definitions, M2M Value Chains, IoT Value Chains, An example for an Industrial Structure **M2M to IoT - An Architectural Overview: B**uilding an architecture, Main design principles and needed capabilities, An IoT architecture outline	**8 Hrs**
MODULE 3: Understanding XaaS Devices and gateways, Data management, Business processes in IoT, Everything as a Service(XaaS), Knowledge Management	**8 Hrs**
MODULE 4: IoT Reference Architecture Reference Architecture: ITU-T, Reference Models: IoT Domain Model, Information Model, Functional Model, Communication model, Safety, Privacy, Trust, Security Model	**8 Hrs**
MODULE 5: IoT Real time Applications Asset Management, Hazardous Goods Management, Other real time applications	**7 Hrs**

Question Paper Pattern:

- **Each full question consists of 20 marks.**
- **Questions are set covering all the topics under each module**

TextBooks

1. Jan Holler, VlasiosTsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle, "From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence",1st Edition,AcademicPress, 2014.

Reference Books

1. Vijay Madisetti and ArshdeepBahga, "Internet of Things (A Hands-on-Approach)", 1st Edition, VPT, 2014.
2. Francis daCosta, "Rethinking the Internet of Things: A Scalable Approach to Connecting Everything", 1st Edition, Apress Publications, 2013

COURSE OUTCOMES (CO)

CO1: Study the evolution of IoT towards global context

CO2: Understand the architecture of IoT and the underlying technology

CO3: Analyze the implications of IoT with real time applications

CO4: Apply the state of the art architecture of IoT to be deployed in real time world

COURSE OUTCOMES MAPPING WITH PROGRAM OUTCOMES:

Course Outcomes(CO)	Mapping with Program Outcomes(PO)
CO1	PO3, PO5
CO2	PO3, PO5, PO8, PO10
CO3	PO3, PO6, PO8
CO4	PO5, PO6, PO10, PO12

V SEMESTER

BLOCKCHAIN TECHNOLOGY

Course code:				**18MCA553**		**CIE Marks:**	**50**
Number of Lecture Hours per week:				**03**		**SEE Marks:**	**50**
Total number of Lecture Hours:				**39**		**SEE Hours:**	**3 Hrs**
Lecture (L):	**3**	**Practicals (P):**	**0**	**Tutorial (T):**	**0**	**Total Credits:**	**3**

COURSE LEARNING OBJECTIVES (CLO)

- **Designed to provide the conceptual understanding of the blockchain**
- **Learn the working technology of blockchain**
- **Understand the application scenarios of blockchain**
- **Implement blockchain in Ethereum technology**

MODULES	TEACHING HOURS
MODULE 1: Basis of Blockchain Technology Introduction to Blockchain, growth – Definition – Elements of Blockchain, Tiers, Types, Consensus, Decentralization: Methods of Decentralization, Routes to decentralization, Blockchain and full ecosystem decentralization	**8 Hrs**
MODULE 2: Blockchain Mining Blockchain: The structure of block, The structure of block header, genesis block – Mining: Tasks, Rewards, Proof of Work, Mining Algorithm, Mining Systems: CPU, GPU, FGPA, ASIC- Mining Pools	**8 Hrs**
MODULE 3: Use case - Financial Markets and Smart Contracts Trading, Exchanges, Trade Lifecycle, order anticipators, Market, Manipulation**,** Smart Contracts: Templates, Smart Oracles, Deploying smart contracts in Blockchain	**8 Hrs**
MODULE 4: Generic Use Cases BlockChain as Evidences – Digital Art - BlockChain Health –Blockchain Government	**8 Hrs**
MODULE 5: Technology on Ethereum Ethereum blockchain, Ethereum network: mainnet, testnet, private net, components of Ethereum ecosystem, Ethereum Virtual Machine	**7 Hrs**

Question Paper Pattern:

- **Each full question consists of 20 marks.**
- **Questions are set covering all the topics under each module**

TextBooks

1. Mastering Blockchain, by Imran Bashir, II edition Packt Publications
2. BlockChain: Blueprint for a new economy, by Melanie Swan O'Reilly Publications

Reference Books

1. "BlockChain: A Beginners Guide", Authors: SherminVoshmgir, Valentin Kalinov Publisher: https://blockchainhub.net/
2. "Cryptocurrency and Bitcoin Technologies", Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, Steven Goldfeder published by Princeton University press 2016

COURSE OUTCOMES (CO)

CO1: Understand the structure and underlying technology of blockchain

CO2: Analyze the application scenarios of blockchain

CO3: Apply the blockchain technology to build a blockchain system

COURSE OUTCOMES MAPPING WITH PROGRAM OUTCOMES:

Course Outcomes(CO)	Mapping with Program Outcomes(PO)
CO1	PO1,PO3
CO2	PO3, PO7, PO10
CO3	PO3, PO4, PO5, PO7, PO10

V SEMESTER
INDUSTRY INTERNSHIP

Course code:				**18MCAI59**	**CIE Marks:**		**50**
Number of Lecture Hours per week:				**-**	**SEE Marks:**		**50**
Total number of Lecture Hours:				**-**	**SEE Hours:**		**3**
Lecture (L):	**0**	**Practicals (P):**	**0**	**Tutorial (T):**	**0**	**Total Credits:**	**05**

Internship - Guidelines

- The students should undergo an internship for 4 weeks during the vacation soon after the 3rd semester SEE.
- The internship shall be carried out in an Industry/R&D labs or at Institution.
- The student should submit the internship report and make the presentation to the internal panel.
- The internal panel will evaluate the internship work for 50 Marks.
- SEE will be conducted for Internship and will be evaluated for 50 Marks

Course Outcomes	
CO1	**Apply domain knowledge in proposing solution for IT problem**
CO2	**Develop/implement the design with appropriate techniques and tools to deliver the solution.**
CO3	**Work in independently or in collaborative environment**
CO4	**Develop communications skills, make presentations and prepare technical document**

VI SEMESTER

TECHNICAL SEMINAR

Course Code				**18MCAS61**		**CIE Marks**	**50**
Number of Practical Hours/Week				**-**		**SEE Marks**	**-**
Total Number of Lecture Hours				**-**		**SEE Hours**	**03**
Lecture (L):	**-**	**Practicals (P):**	-	**Tutorial (T):**	**-**	**Total Credits:**	**2**

Course Learning Objectives:

- Students will present technical seminar by learning new technologies

Technical Seminar

Seminar Guidelines

- Selection of topic/area : Select a paper according to the specialization of students. Papers from any reputed journals or latest technology shall be selected.
- Obtain the approval from the guide for the selected topic.
  Study of topic: Students are informed to acquire a thorough knowledge on the subject by referring back papers and reference books (These may be included as references at the end of the paper) on the corresponding area.
- Preparation of slides for presentation: Slides may be presented in MS power point. Time allowed for presentation is 20 minutes for presentation and 5 minutes for discussions. So, number of slides may be around 20 - 25 to adhere the time limit.

  Organization of slides:

  The first slide will be a title page showing the title, name of student (presenter), USN. and Semester. 2nd page will contain overview of the seminar
- Successive pages will contain

  Objectives of the paper

Introduction,

Body of the paper includes system dynamics, methodology, graphs, block diagrams etc. arranged in a logical sequence depending on the problem.

Results and discussions

Conclusion

- Last page will contain references and bibliography. References must be presented in IEEE format
- Each slide consists of 4 or 5 lines with enough space between lines.
- All equations must be typed using equation editor (available with MS office/other office suite)
- Each slide will have a title and each figure have a caption.

. The internal panel will evaluate the seminar presentation for 50 Marks.

COURSE OUTCOME:

CO1: Students will present a seminar on any new technology or any research topic.

COURSE OUTCOMES MAPPING WITH PROGRAM OUTCOMES:

Course Outcomes(CO)	**Mapping with Program Outcomes(PO)**
CO1	**PO1,PO4,PO5,PO8,PO11**

VI SEMESTER

PROJECT WORK

Course code:				**18MCAP62**	**CIE Marks:**		**50**
Number of Lecture Hours per week:				**-**	**SEE Marks:**		**200**
Total number of Lecture Hours:				**-**	**SEE Hours:**		**3**
Lecture (L):	**0**	**Practicals (P):**	**0**	**Tutorial (T):**	**0**	**Total Credits:**	**20**

Synopsis

- The Synopsis of the project must be submitted before the third week of 4th semester.
- The synopsis of the project must include:

  a) Problem formulation and literature survey.

  b) Details of the required tools and technologies for the development of project.

  c) Write up shall not exceed 15 pages.
- Internal assessment for synopsis presentation and evaluation of the synopsis by the internal panel /guide is for 100 marks.

Dissertation:

- The project shall be carried out in the same institution or in industry/R&D labs, based on software tools and technologies learnt in MCA course/internship for minimum period of 16 weeks.
- Internal assessment shall be evaluated by the internal panel/guide for **50** marks. For continuous evaluation of project work by the internal examiner/guide with progress reports is for 10 marks each. (3 progress reports x 10 marks= 30)
- Final presentation for the entire project is evaluated for 20 marks by the project Guide.
- The internal examiners (Project Guide with at least 3 years of experience) and the

external examiners shall be appointed by the authorities of the college for the final evaluation of the project.

- Internal and external examiners shall carry out the evaluation of Dissertation report for **100** marks individually. The average of the marks allotted by the internal and external examiners shall be the final marks for the project Dissertation report evaluation.
- The project presentation and Viva-voce examination shall be evaluated jointly by both the internal and external examiners for **100** marks.
- The student shall publish the project outcome in the reputed journals.

COURSE OUTCOMES MAPPING WITH PROGRAM OUTCOMES:

Course Outcomes(COs)	**Mapping with Program Outcomes(POs)**
CO1	**PO1,PO2,PO3,PO4**
CO2	**PO1,PO2,PO3,PO4,PO5,PO7,PO8**
CO3	**PO4,PO5,PO7,PO8,PO10,PO11**
CO4	**PO4,PO5,PO7,PO8,PO9,PO10,PO11,PO12**