

TUMKUR UNIVERSITY

TUMKUR

**Scheme and Syllabus of MCA Course
W.E.F 2017-18 as per CBCS Guidelines**



**Department of Studies and Research in
Computer Applications (MCA)
Tumkur University, B H Road
Tumakuru.**

COURSE MATRIX SEMESTER WISE

SEMESTER-I

Sl.No	Course Code	Subject Code	Subject Title	Teaching Hours/Week			Credits	Exam Hours	CIE	SSE	Total
				L	T	Lab					
1	17MCA11	CPT 1.1	Problem Solving Using C	03	02	-	04	3	20	80	100
2	17MCA12	CPT 1.2	Unix Programming	04	-	-	04	3	20	80	100
3	17MCA13	CPT 1.3	Web Technologies	04	-	-	04	3	20	80	100
4	17MCA14	SPT 1.4	Fundamentals of Computer Organization	04	-	-	04	3	20	80	100
5	17MCA15	SPT 1.5	Discrete Mathematical Structures	04	-	-	04	3	20	80	100
6	17MCA16	CPP 1.6	C Programming Lab	-	-	04	02	3	10	40	50
7	17MCA17	CPP1.7	Unix Programming Lab	-	01	02	02	3	10	40	50
8	17MCA18	SPP 1.8	Web Programming Lab	-	01	02	02	3	10	40	50
Total Credits							26	Total Marks		650	
Total Contact Hours /Week							31				

CPT: Core Paper Theory,
SPT: Special paper Theory,
OEPT: Open Elective Paper Theory,
OEPP: Open Elective Paper Practical.

SEMESTER-II

Sl.No	Course Code	Subject Code	Subject Title	Teaching Hours/Week			Credits	Exam Hours	CIE	SSE	Total
				L	T	Lab					
1	17MCA21	CPT 2.1	Operating Systems	04	-	-	04	3	20	80	100
2	17MCA22	CPT 2.2	Object Oriented Programming with C++	03	02	-	04	3	20	80	100
3	17MCA23	CPT 2.3	Data Base Management Systems	04	-	-	04	3	20	80	100
4	17MCA24	OEPT 2.4	Data Structures using C++	04	-	-	04	3	20	80	100
5	17MCA25	SPT 2.5	Computer Networks	04	-	-	04	3	20	80	100
6	17MCA26	CPP 2.6	OS Lab	-	01	02	02	3	10	40	50
7	17MCA27	CPP 2.7	DBMS Lab	-	01	02	02	3	10	40	50
8	17MCA28	OEPP 2.8	Data Structures Lab Using C++	-	01	02	02	3	10	40	50
Total Credits							26	Total Marks			650
Total Contact Hours /Week							30				

SEMESTER-III

Sl.No	Course Code	Subject Code	Subject Title	Teaching Hours/Week			Credits	Exam Hours	CIE	SSE	Total
				L	T	Lab					
1	17MCA31	CPT 3.1	Programming Using Java	4	-	-	4	3	20	80	100
2	17MCA32	CPT 3.2	Python programming	4	-	-	4	3	20	80	100
3	17MCA33	CPT 3.3	Software Engineering	4	-	-	4	3	20	80	100
4	17MCA34	SPT 3.4	Computer Graphics	4	-	-	4	3	20	80	100
5	17MCA35	OET 3.5	Analysis & Design of Algorithms	4	-		4	3	20	80	100
6	17MCA36	CPP 3.6	Java Programming Lab	-	-	4	2	3	10	40	50
7	17MCA37	CPP3.7	Python Lab	-	-	4	2	3	10	40	50
8	17MCA38	OEPP3.8	ADA Lab	-	-	4	2	3	10	40	50
Total Credits							26	Total Marks			650
Total Contact Hours /Week							30				

SEMESTER-IV

Sl.No	Course Code	Subject Code	Subject Title	Teaching Hours/Week			Credits	Exam Hours	CIE	SSE	Total
				L	T	Lab					
1	17MCA41	CPT 4.1	Advanced Java	04	-	-	04	3	20	80	100
2	17MCA42	CPT4.2	Advanced Web Programming	04	-	-	04	3	20	80	100
3	17MCA43	CPT 4.3	Mobile Applications	04	-	-	04	3	20	80	100
4	17MCA44	SPT 4.4	Elective-I	04	-	-	04	3	20	80	100
5	17MCA45	SPT 4.5	Elective-II	04	-	-	04	3	20	80	100
6	17MCA46	CPP 4.6	Adv. Java Lab	-	-	04	02	3	10	40	50
7	17MCA47	CPP4.7	Adv. Web Lab	-	-	04	02	3	10	40	50
8	17MCA48	SPP4.8	Mini Project-I	-	-	04	02	3	10	40	50
Total Credits							26	Total Marks			650
Total Contact Hours /Week							30				
Elective-I											
						Title					
SPT 4.4		17MCA441			Basics of MIS & e-Commerce						
		17MCA442			ERP						
		17MCA443			Introduction to AI						
Elective-II											
						Title					
SPT 4.5		17MCA451			Principles of User Interface Design						
		17MCA452			Multimedia Communications						
		17MCA453			Internet of Things(IoT)						

SEMESTER-V

Sl. No	Course code	Subject Code	Subject Title	Teaching Hours/Week			Credits	Exam Hours	CIE	SSE	Total
				L	T	Lab					
1	17MCA51	CPT 5.1	Object-Oriented Modeling and Design Patterns	04	-	-	04	3	20	80	100
2	17MCA52	CPT 5.2	Data Mining	04	-	-	04	3	20	80	100
3	17MCA53	SPT 5.3	Computational Intelligence	04	-	-	04	3	20	80	100
4	17MCA54	SPT 5.4	Fuzzy Logic and Applications	04	-	-	04	3	20	80	100
5	17MCA55	SPT 5.5	Digital Image Processing	04	-	-	04	3	20	80	100
6	17MCA56	CPP 5.6	Data Mining Lab	-	-	04	02	3	10	40	50
7	17MCA57	CPP 5.7	Software Design Lab	-	-	04	02	3	10	40	50
8	17MCA58	SPP 5.8	Mini Project –II	-	-	04	02	3	10	40	50
Total Credits							26	Total Marks			650
Total Contact Hours /Week							32				

SEMESTER-VI

Sl.No	Course code	Subject Code	Subject Title	Teaching Hours/Week			Credits	Exam Hours	CIE	SSE	Total
				L	T	Lab					
1.	17MCA61	CPT 6.1	E-Governance	4	-	-	4	03	20	80	100
2.	17MCA62	CPT 6.2	Parallel Algorithms	4	-	-	4	03	20	80	100
3.	17MCA63	CPT 6.3	Advanced Database Management Systems	4		-	4	03	20	80	100
4.	17MCA64	SPT 6.4	Dissertation/ Major Project	-	-	28	14	03	100	250	350
Total Credits							26	Total Marks		650	
Total Contact Hours /Week							40				

SEMESTER-I

Subject Name: Problem Solving Using C			
Subject Code:	17MCA11	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	04
Detailed Syllabus			
<p>Module-1: 12 Hrs</p> <p>Algorithms and Flowcharts The meaning of algorithms, Flowcharts and their need, Writing algorithms and drawing flowcharts for simple exercises like finding biggest of three numbers, to find roots of given quadratic equation, to find the biggest and smallest of given set of numbers and such other simple examples</p> <p>Introduction to programming in C, Constants, Variables and Data Types Character set, C tokens, keywords & identifiers, structure of C program, executing a C program. Constants, variables, data types, declaration of variables, declaration of storage classes, assigning values to variables defining symbolic constants, declaring a variable as constant, declaring a variable as volatile, overflow and underflow of data.</p> <p>Operators and Expressions Arithmetic operators, relational operators, logical operators, assignment operator, increment and decrement operator, conditional operator, bitwise operators, comma operator, special operators, arithmetic expressions, evaluation of expressions, precedence of arithmetic operators, type conversions in expressions, operator precedence and associability, mathematical functions</p> <p>Module -2: 10 Hrs</p> <p>Managing Input and Output Operations The scanf() & printf() functions for input and output operations, reading a character, writing a character, (the getchar() & putchar() functions) , the address operator(&), formatted input and output using format specifiers, Writing simple complete C programs.</p> <p>Control Statements Decision making with if statement, simple if statement, the if..else statement, nesting of if..else statements, the else..if ladder, the switch statement, the ? : operator, the goto statement, the break statement, programming examples</p>			

The while statement, the do. While statement, the for statement, nested loops, jumps in loops, the continue statement, programming examples

Module -3: 10 Hrs

Arrays

The meaning of an array, one dimensional and two dimensional arrays, declaration and initialization of arrays, reading , writing and manipulation of above types of arrays, multidimensional arrays, dynamic arrays, programming examples.

Character Arrays and Strings

Declaring and initialing string variables, reading string from terminal, writing string to screen, arithmetic operations on characters, putting strings together, comparison of two strings, string handling functions, table of strings, other features of strings, programming examples.

Module – 4: 10 Hrs

User Defined Functions

Need for user defined functions, a multi-function program, elements of User defined functions, defining functions, return values and their types, function calls, function declaration, category of functions, no arguments and no return values, arguments but no return values, arguments with return values, no arguments with return value, functions that return multiple values, nesting of functions, recursion, passing arrays to functions, passing string to functions, programming examples.

Structures and Unions

Defining a structure, declaring structure variables, accessing structure members, structure initialization, copying and comparing structure variables, operations on individual members, array of structures, structures within structures, structures and functions, Unions, size of structures, programming examples

Module -5: 8Hrs

Pointers

Understanding pointers, accessing the address space of a variable, declaring and initialization pointer variables, accessing a variable through its pointer, chain of pointers, pointer expressions, pointers and arrays, pointer and character strings, array of pointers, pointer as function arguments, functions returning pointers, pointers to functions, pointers and structures, programming examples

Text Books:

1. Behrouz A Forouzan and Richard F Gilberg, Structured Programming Approach in C, 2nd Edition, Thomson, 2001.
2. V Rajaraman, Computer Programming in C, Prentice Hall India, 2000. Chapters: 1.1,1.3, 2.1, 2.3, 3.1, 3.2, 3.3

Subject Name: UNIX Programming			
Subject Code:	17MCA12	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	04
Detailed Syllabus			
<p>Module -1: 10 Hrs Introduction of UNIX and Shell: Introduction, History, Architecture, Experience the Unix environment, Basic commands ls, cat, cal, date, calendar, who, printf, tty, sty, uname, passwd, echo, tput, bc, script, spell and ispell, Introduction to Shell Scripting, Shell Scripts, read, Command Line Arguments, Exit Status of a Command, The Logical Operators && and , exit, if, and case conditions, expr, sleep and wait, while, until, for, \$, @, redirection. The here document, set, trap, Sample Validation and Data Entry Scripts.</p> <p>Module -2: 10 Hours UNIX File System: The file, what's in a filename? The parent-child relationship, pwd, the Home directory, absolute pathnames, using absolute pathnames for a command, cd, mkdir, rmdir, Relative pathnames, The UNIX file system. Basic File Attributes: Is - l, the -d option, File Permissions, chmod, Security and File Permission, users and groups, security level, changing permission, user masks, changing ownership and group, File Attributes, More file attributes: hard link, symbolic link, umask.</p> <p>Module -3: 10 Hours Simple Filters: Pr, head, tail, cut, paste, sort, uniq, tr commands, Filters using Regular Expression : grep & sed grep, Regular Expression, egrep, fgrep, sed instruction, Line Addressing, Inserting and Changing Text, Context addressing, writing selected lines to a file, the -f option, Substitution, Properties of Regular Expressions Context addressing, writing selected lines to a file, the -f option, Substitution, Properties of Regular Expressions.</p> <p>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-in variables, Arrays, Functions, Interface with the Shell, Control Flow. The sh command, export, cd, the Command, expr, Conditional Parameter Substitution, Merging Streams, Shell Functions, eval, Exec Statement and</p>			

Examples.

Module -5: 10 Hours

privileges, startup & shutdown, managing disk space, cpio, tar, Customizing the Environment : System Variables, profile, sty, PWD, Aliases, Command History, On-line Command Editing. Advanced System Administration: Case Study: emacs editor and any one distribution of Linux.

Text Books:

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

Reference Books:

1. "Unix Shell Programming", Yashwant Kanetkar, Beginning Shell Scripting", Eric Foster-Johnson, John CWelch, Micah Anderson, Wrox publicationS.
2. UNIX: Concepts and Applications, Sumitabha Das, Tata Mc Graw Hill,
3. "Introduction to UNIX" by M.G.Venkatesh Murthy.

Subject Name: Web Technologies			
Subject Code:	17MCA13	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	04
Detailed Syllabus			
Module -1: 10 Hours			
Web Fundamentals Internet, WWW, Web Browsers and Web Servers, URLs, MIME, HTTP, Security, the Web Programmers Toolbox. Evolution of the Web, Peak into the History of the Web, Internet Applications, Networks, TCP/IP, Higher Level Protocols, Important Components of the Web, Web Search Engines, Application Servers			
Module -2: 10 Hours			
Introduction to XHTML and CSS Basic syntax, Standard structure, Basic text markup, Images, Hypertext Links. Lists, Tables, Forms, Frames, syntactic differences between HTML and XHTML.			
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 <div> 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: 10 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, XML documents Displaying with CSS, XSLT style sheets, XML processors, Webservices.			

Perl and CGI Programming

Origins and uses of Perl, Scalars and their operations, Assignment statements and simple input and output, Control statements, Fundamentals of arrays, Hashes, References, Functions, Pattern matching, File input and output; Examples.

Using Perl for CGI Programming: The Common Gateway Interface; CGI linkage; Query string format; CGI.pm module; A survey example; Cookies.

Text Books:

1. Robert W. Sebesta : Programming the World Wide Web, 4th Edition, Pearson education, 2012. (**Chapters 1, 2, 3, 4, 5, 6, 7, 8, 9**)
2. M. Srinivasan: Web Programming Building Internet Applications, 3rd Edition, Wiley India, 2009. (**Chapter 1**)

Reference Books:

1. Jeffrey C Jackson : Web Technologies – A Computer Science Perspective Pearson Education, 7 Impressions, 2012. Chris Bates: Web Technology Theory and Practice, Pearson Education, 2012.
2. Internet Technology and Web Design, Instructional Software Research and Development (ISR D) Group, Tata McGraw Hill, 2011 Raj Kamal Internet and Web Technologies, McGraw Hill Education.

Subject Name: Fundamentals of Computer Organization			
Subject Code:	17MCA14	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	04
Detailed Syllabus			
<p>Module -1: 10 Hrs Number Systems and Logic Gates Counting in Decimal and Binary, Place Value, Binary to Decimal Conversion, Decimal to Binary Conversion, Hexadecimal Numbers, Octal Numbers, Bits, Bytes, Nibbles, and Word Size, The AND Gate, The OR gate, The inverter and Buffer, The NAND gate, the NOR Gate, The exclusive OR gate, The Exclusive NOR Gate, The NAND Gate as a universal Gate, Gates with More than two inputs, Using Inverters to convert gates.</p> <p>Combining Logic Gates Constructing Circuits from Boolean Expressions, Drawing a circuit from a Maxterm Boolean Expression, Truth Tables and Boolean Expressions, Sample Problem, Simplifying Boolean Expressions, Karnaugh Maps, Karnaugh Maps with three variables, Karnaugh Maps with four variables, more Karnaugh Maps, using Demorgan's Theorems</p> <p>Module -2: 10 Hrs Codes and Arithmetic Circuits The 8421 BCD Code, The Excess-3 Code, The Gray Code, The ASCII Code. Binary Addition, Half Adders, Full Adders, Three Bit Adders, Binary Subtraction, Parallel Subtractors, IC Adders, Binary Multiplication, Binary Multipliers, 2s Complement Notation, Addition and Subtraction, 2s Complement adders/subtractor</p> <p>Basic structure of computers Computer types, Functional units (Input unit, Memory unit, ALU, Output unit, Control unit), Basic operational concepts, Bus structures</p> <p>Module -3: 12 Hrs Machine Instructions & Programs Memory locations and addresses (Byte addressability, BIG-ENDIAN & LITTLE-ENDIAN Assignments, Word alignment, Accessing numbers, characters, & character strings), Memory operations, Instruction and instruction sequencing (Register transfer notation, Assembly language notation, Basic instruction types, Instruction execution & straight-line sequencing, Branching, Condition codes, Generating memory addresses), Addressing modes (Implementation of variables and constants, Indirection and pointers, Indexing & arrays, Relative addressing, Additional modes), Basic Input/Output operations, Stacks and queues, Subroutines (Subroutine nesting & the Processor stack, Parameter passing, The stack frame)</p> <p>Module-4: 08 Hrs Input/Output Organization</p>			

Handling multiple devices, Controlling device requests, Exceptions), Direct memory access (Bus arbitration), Buses (Synchronous bus, Asynchronous bus), Interface circuits (Serial port)

Module-5 : 10 Hrs

The Memory System

Some basic Concepts, Semiconductor RAM memories (Internal organization of memory chips, Static memories, Asynchronous DRAMs, Synchronous DRAMs, Memory system considerations, Rambus memory), Read Only Memories (ROM, PROM, EPROM, EEPROM, Flash memory), Speed, size & Cost, Cache memories (Mapping functions), Virtual memories (Address translation), Types of Intel processors.

Text Books:

1. Tokheim: Digital Electronics Principles and Applications, McGraw Hill, 6th Edition, 2004. (Chapters: 2.1 to 2.4, 2.6 to 2.8, 3.1 to 3.10, 4.1 to 4.9, 4.16, 6.1 to 6.4, 7.1 to 7.4, 8.4, 10.1 to 10.11)
2. Carl Hamacher, Z Varnesic and S Zaky: Computer Organization, 5th Edition, McGraw Hill, 2002. (Chapters: 1.1 to 1.4, 2.2 to 2.5, 2.7 to 2.9, 4.1, 4.2, 4.4 to 4.6, 5.1 to 5.5, 5.7,11.3)

Reference Books:

1. M Morris Mano: Digital Logic and Computer Design, 10th Edition, Pearson, 2008.
2. Morris Mano, and Charles R Kime: Logic and Computer Design Fundamentals, 2nd Edition, Pearson Education, 2001.
3. Mostafa Abd-El-Barr & Hesham El-Rewini, Fundamentals of Computer Organization and Architecture, (2005) Wiley

Subject Name: Discrete Mathematical Structures			
Subject Code:	17MCA15	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	04
Detailed Syllabus			
<p>Module -1: 10 Hours Logic: Propositional logic, equivalences, predicates and quantifiers, rules of inference, introduction to proofs, proof methods.</p> <p>Module -2: 10 Hours Sets, Functions and Relations: Sets, set operations, Functions, Relations, equivalence relations and partial ordering.</p> <p>Module -3: 10 Hours Counting: Basics of counting, the pigeonhole principle, permutations and combinations, Binomial Co-efficient, recurrence relations.</p> <p>Module -4: 10 Hours Probability: Introduction to probability, axioms of probability, independence and conditional probability, inclusion-exclusion principle.</p> <p>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, planar graphs, graph coloring.</p> <p>Text Books:</p> <ol style="list-style-type: none"> 1. Discrete Mathematics and its Applications by Kenneth H Rosen, 7th edition, (Indian adaptation by Kamala Krithivasan), Tata McGraw Hill, 2011. 2. Discrete and Combinatorial Mathematics: An Applied Introduction by Ralph P. Grimaldi and B V Ramana (Indian corrupted adaptation), 5th edition, Pearson, 2011. 			

Subject Name: C Programming Laboratory			
Subject Code:	17MCA16	CIE Marks	10
No of Hours/Week:	4	SEE Marks	40
Total Hours	50	Credits	02
Detailed Syllabus			
Part A			
<ol style="list-style-type: none"> Exercise on writing simple C Programs (like find the area of a triangle, Calculate the simple interest with appropriate validations). Exercise on writing simple C programs using <i>if else</i> statements (like Find biggest of three numbers). Exercise on writing simple C programs using <i>switch</i> statement (like provide multiple operations, addition, subtraction, etc.). Exercises on writing C programs using loop structures (like find the sum of first 'N' natural numbers). Exercise on writing C programs using functions (like Generate and print first n Fibonacci numbers). Exercise on writing C programs using recursive <i>function</i> (like Find the factorial of a given number). Exercise on writing C programs using characters <i>and string handling</i> (like Read a sentence and count the number of vowels and constants). To check whether a given number is prime or not. To find GCD and LCM of two integers. And to check whether a given number is Armstrong or not. 			
Part B			
<ol style="list-style-type: none"> Exercise on writing C programs using <i>arrays</i> (like sorting a list of integers, bubble sorting and selection sorting). Exercise on writing C programs using <i>arrays</i> and <i>functions</i> (like searching a number from a list of integers using binary search). Exercise on writing C programs using <i>two-dimensional arrays</i> (like addition matrices and subtraction of matrices after checking for compatibility) Exercise on writing C programs using <i>structures</i> (linear search, sorting of array of structures). Exercise on writing C programs using <i>structures</i> and <i>functions</i> (like compute the sum of two complex numbers, passing a structure to a function). Exercise on writing C programs using <i>pointers</i> (like swap two numbers using Function). Exercise on evaluating a polynomial using Horner's method. Note: Students are required to execute one question from Part A and one from Part B 			

Subject Name: UNIX Programming Lab			
Subject Code:	17MCA17	CIE Marks	10
No of Hours/Week:	4	SEE Marks	40
Total Hours	50	Credits	2
Detailed Syllabus			
<p>Laboratory Experiments:</p> <p>A. Explore the Unix environment.</p> <p>B. Explore vi editor with vim tutor. Perform the following operations using vi editor, but not limited to:</p> <ol style="list-style-type: none"> 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. set line numbers 9. search for a pattern 10. move forward and backward <p>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.</p> <p>1b. Write a shell script that accepts a pathname and creates all the components in that path name 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</p> <p>2a. Write a shell script that accepts two file names as arguments, checks if the permissions for these files are identical and if the permissions are identical, output common permissions and otherwise output each file name followed by its permissions.</p> <p>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.</p> <p>3a. Create a script file called file-properties that reads a file name entered and outputs its Properties</p> <p>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 must lock the keyword until a matching password 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.</p> <p>4a. Write a shell script that accept one or more filenames as argument and convert all of them</p>			

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 at least 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 a shell script that reports the logging in of a specified user within one minute after he/she log in. The script automatically terminate if specified user does not log in during a specified period of time.

9a. Write a shell script that accept the file name, 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 40 columns. Thus any line that exceeds characters must be broken after th a "\" is to be appended as the indication of folding and the 40 , processing is to be continued with the residue. The input is to be supplied through a text file created by the user.

10a. Write an awk script that accepts date argument in the form of dd-mm-yy and displays 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
Computer Science 43
Mechanical 65
Civil 98
Computer Science 64

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

If basic salary is < 10000 then HRA=15% of basic & DA=45% of basic If basic salary 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 all 1-11 questions.

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

Subject Name: Web Programming Laboratory			
Subject Code:	17MCA18	CIE Marks	10
No of Hours/Week:	4	SEE Marks	40
Total Hours	50	Credits	2
Detailed Syllabus			
<p><i>Laboratory Experiments:</i></p> <p>1. Create an XHTML page that provides information about your department. Your XHTML page must use the following tags:</p> <ol style="list-style-type: none"> Text Formatting tags Horizontal rule Meta element Links Images Tables <p>(If needed use additional tags).</p> <p>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.</p> <p>3. 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.</p> <p>Sampleoutput form</p>			

Welcome to Millennium Gymnastics Booster Club Popcorn Sales

Buyer's Name:

Street Address:

City, State, Zip:

Product Name	Price	Quantity
Unpopped Popcorn (1 lb.)	\$3.00	<input type="text"/>
Caramel Popcorn (2 lb. canister)	\$3.50	<input type="text"/>
Caramel Nut Popcorn (2 lb. canister)	\$4.50	<input type="text"/>
Toffee Nut Popcorn (2 lb. canister)	\$5.00	<input type="text"/>

Payment Method:

Visa Master Card Discover Check

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. Develop, test and validate an XHTML document that has checkboxes for apple (59cents 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.

6. a) Develop and demonstrate, a XHTML 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 good-looking and effective.

b)Modifythe above program to get the current semester also(restricted to be anumber from 1 to 6

7. 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 moved from the top stacking position, it returns to its original position rather than to the bottom.

8. 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.

A SIMPLE CLACULATOR

Number 1 =

Number 2 =

Result =

Modify your program to make HTML document as eye-catching using CSS.

9. a. 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.

b. Create an XSLT style sheet for one student element of the above document and use it to create a display of that element.

10. Write a Perl program to process a file which contains English words, where each word is separated from the next word on a line by one space. The file is specified on the command line. The output of your program is a table in which the first column has unique words from the input file and second the second column has the number of times the word appeared in the file; no word can appear twice in the table. Use two arrays to store the table, one for the words and one for the frequency values.

Modify the program to count number of characters, words, and lines in the specified file

11. A file contains lines of employee data, where each line has name:age:department code:salary. Write a Perl program to generate the following output:

1. The names of all the employee whose names end with "son"
2. Percentage of employees under 40 years old
3. Average salary of employees under 40 years old
4. An alphabetical list of employees who are under 40 years old and who have salaries more than \$40,000.

12. Write a Perl program to accept the Username and display a greeting message randomly chosen from a list of 4 greeting messages. Modify this Perl program to count the number of visitors visiting this web page and display that number of times each visitor is visited.
13. Write an XHTML document to create a form with the following capabilities:
 - a. A text widget to collect the user's name
 - b. B four checkboxes, one each for the following items
 - I. Four 100-watt light bulbs for \$2.39
 - II. Eight 100-watt light bulbs for \$4.29
 - III. Four 100-watt long-life light bulbs for \$3.95
 - IV. Eight 100-watt long-life light bulbs for \$7.49
 - c. A collection of three radio buttons they are labeled as follows:
 - I. Visa
 - II. MasterCard
 - III. Discover

Write a Perl CGI program that computes the total cost of the ordered light bulbs by a customer. The program must inform the buyer of exactly what was ordered in a table.

14. Write a CGI-Perl program to use a cookie to remember the day of the last login from a user and display it when run. Modify this program to return the number of months, days, hours, and minutes since last visit by the current client. Your display should include current visit day, date and time, last visit day, date, and time, & elapsed time.

The sample out is as follows: User

name: XXXXXXXXXXXX

Current visit time : Friday 05/08/2016, 9:30:30

3:20:00 PM Elapsed time : 18:10:30

Note 1: In the practical Examination each student has to pick one question from a lot of all 14 questions.

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

SEMESTER-II

Subject Name: Operating Systems			
Subject Code:	17MCA21	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	04

Detailed Syllabus

Module -1: 10 Hours

Computer and Operating Systems Structure

Basic Elements, Processor Registers, Instruction Execution, The Memory Hierarchy, Cache Memory, I/O Communication Techniques, Introduction to Operating System, Mainframe Systems, Desktop Systems, Multiprocessor Systems, Distributed Systems, Clustered Systems, Real - Time Systems, Handheld Systems, Feature Migration, Computing Environments. **System Structures:**

System Components, Operating – System Services, System Calls, System Programs, System Structure, Virtual Machines, System Design and Implementation, System Generation

Module -2: 10 Hours

Process Management and Mutual Execution

Process, Process States, Process Description, Process Control, Execution of the Operating System, Security Issues, Processes and Threads, Symmetric Multiprocessing(SMP), Micro kernels, CPU Scheduler and Scheduling. Principles of Concurrency, Mutual Exclusion: Hardware Support, Semaphores , Monitors , Message Passing, Readers/Writes Problem.

Module -3: 10 Hours

Deadlock and Memory Management

Principles of Deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, An Integrated Deadlock Strategy, Dining Philosophers Problem Memory Management: Swapping, Contiguous Memory Allocation, Paging, Segmentation, Segmentation with Paging, Demand Paging, Process Creation, Page Replacement, Allocation of Frames, Thrashing

Module -4: 10 Hours

File System and Secondary Storage

File Concept, Access Methods, Directory Structure, File System Mounting, File Sharing, Protection, File – System Structure, File – System Implementation, Directory Implementation, Allocation Methods, Free–Space Management, Disk Structure, Disk Scheduling, Disk Management.

Module -5: 10 Hours

Computer Security and Case study of Linux Operating system

The Security Problem, User Authentication, Program Threats, System Threats. Linux System Linux history , Design Principles, Kernel modules, Process , management,

communications

Text Books:

1. Silberschatz, Galvin, Gagne, "Operating System Concepts" John Wiley, Sixth Edition, 2004
2. William Stallings, "Operating System Internals and Design Principles" Pearson, 6th edition, 2012

Reference Books:

1. Chakraborty, "Operating Systems" Jaico Publishing House, 2011.
2. Dhananjay M. Dhamdhere, "Operating Systems – A Concept – Based Approach", Tata McGraw Hill, 3rd Edition, 2012.

Subject Name: Object Oriented Programming with C++			
Subject Code:	17MCA22	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	04
Detailed Syllabus			
<p>Module -1: 10 hrs Overview of OOP Object Oriented paradigm, Structured vs. Object Oriented Paradigm. Elements of Object Oriented Programming. C++ Overview Introduction, different data types, Reference Variables in C++. Function Components, argument passing mechanisms, inline functions, function overloading.</p> <p>Module -2: 10 hrs Modular Programming Introduction to Classes and objects: Introduction, Class Specification, Class Objects, access members, defining member functions, constructors, destructors, Static data members, functions, scope resolution operator, Passing objects as arguments, returning objects.</p> <p>Module -3: 10 hrs <i>Classes & Objects continued...</i> Friend functions & classes, array of objects. Operator Overloading: Operator Overloading, Overloading the Arithmetic Operators, Overloading the Increment and the Decrement Operators (Prefix and Postfix) using member and non member functions.</p> <p>Module -4: 10 hrs Inheritance: Introduction to Inheritance, Protected Access Specifier, Deriving by Different Access Specifiers, Different Kinds of Inheritance, Order of invocation of Constructors and Destructors, Passing parameters to base Class Constructors, Granting access, Virtual base classes.</p> <p>Module -5: 10 hrs Virtualfunction, Polymorphism and I/O stream Library Dynamic objects – Pointers to objects, Class members. Virtual Functions and Dynamic Polymorphism: The Need for Virtual Functions, Virtual Functions, The Mechanism of Virtual Functions, Pure Virtual Functions, Early & late binding. IO Stream Library. Templates: Introduction, class templates and function templates with multiple parameters, overloading of templates functions.</p> <p>Text Book: 1. H. Schild: C++ The Complete Reference, 4th Edition, Tata McGraw Hill, 2007.</p>			

Subject Name: Database Management System			
Subject Code:	17MCA23	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	04
Detailed Syllabus			
<p>Module -1: 10 Hours Introduction 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.</p> <p>Module -2: 10 Hours Relational Model Relational Model and Relational Algebra: Relational Model Concepts, RelationalModelConcepts, Relational Model Constraints and Relational Database Schema Update Operations, Transactions and Dealing with Constraint violations, Unary Relational operations, Relational Algebra Operations from Set Theory, Binary Relational Operations, JOIN and DIVISION, Additional Relational Operations, Examples of Queries in Relational Algebra Relational Database Design Using ER-to- Relational Mapping</p> <p>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, SQL Data Types and Schemas, Authorization. Database programming issues and techniques, Embedded SQL.</p> <p>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.</p> <p>Module -5: 10 Hours Transaction Management: Transaction Concept, A Simple Transaction Model, Transaction Atomicity and Durability, Serializability, Transaction Isolation and Atomicity, Transaction</p>			

Implementation of Isolation Levels. Concurrency Control: Lock Based Protocols, Deadlock Handling. Recovery System: Failure Classification, Storage, Recovery and Atomicity, Recovery Algorithm.

Text Books:

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

Reference Books:

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

Subject Name: Data Structures Using C++			
Subject Code:	17MCA24	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	04
Detailed Syllabus			
<p>Module -1: 10 Hours</p> <p>Introduction to Data Structures</p> <p>Information and its meaning: Abstract Data Types, ADT's for Arrays, Stack, Queue, Linked list and Strings. Data Types, Pointers and review of Pointers. Discussion on Arrays and Strings: Array as an ADT, Using One-dimensional Arrays, Implementing One-Dimensional Arrays, Arrays as Parameters, Handling of Character Strings and Character Strings.</p> <p>Module -2: 10 Hours</p> <p>Stack and Recursion</p> <p>Definition and examples , Primitive operations, Example, The stack as an ADT, Representing stacks ,Implementing the pop operation, Testing for exceptional conditions , Implementing the push operations , Examples for infix , postfix, and prefix expressions, Basic definition and Examples ,Program to evaluate a postfix expression ,Converting an expression from infix to postfix, Program to convert an expression from infix to postfix, Applications of Stacks: Expression Evaluations, Recursion etc. Recursion: definition and processes, Factorial function, Multiplication of natural numbers, Fibonacci sequence, Binary search, Properties of recursive definition or algorithm. Binary search, Towers of Hanoi problem.</p> <p>Module -3: 10 Hours</p> <p>Queues and Lists</p> <p>The queue and its sequential representation, the queue as ADT, Insert operation, Priority queue, Array implementation of a priority queue. Linked lists, Inserting and removing nodes from a list, Linked implementations of stacks, getnode and Freenode operations, Linked implementation of queues, Linked list as a data Structure, Example of list operations, Header nodes, Array implementation of lists, Limitations of array implementation, allocating and freeing dynamic variables, Linked lists using dynamic variables, Non integer and non- homogenous lists, Other list structures: Circular lists, Stack as a circular lists, doubly linked lists, Application of Linked Lists: Stacks, Queues, Double-ended Queues, Priority Queues.</p> <p>Module -4: 10 Hours</p> <p>Sorting and Searching</p> <p>Bubble sort, Quick sort, Selection sort, Tree Sorting: Binary Tree Sort, Heap Sort, Insertion Sorts: Simple Insertion, Shell Sort, Address Calculation Sort, Merge and Radix Sort. Basic Search Techniques: Algorithmic Notations, Sequential searching, Searching an ordered table, Indexed sequential search, Binary search, Interpolation search, Tree searching: Inserting into a Binary Search Tree ,Deleting form a binary search tree, Hashing : Resolving hash clashes by open addressing, Choosing a hash</p>			

Binary Trees

Tree traversals, Binary Search Tree and Operations, AVL Tree and Operations, Red-Black Tree, Threaded binary trees and operations.

Text Books:

1. Data Structures Using C and C++ by Yedidyah Langsam and Moshe J. Augenstein and Aaron M Tenanbanum, 2nd Edition, Pearson Education Asia, 2002.

Reference Books:

1. Balaguruswamy: Data Structures Using C, McGraw Hill Education
2. Data Structures and Algorithm Analysis in C, Mark Allen Weiss, 2nd Edition, Pearson Education Aisa, 1997.
3. Richard F Giberg and Behrouz A Forouzan: Data Structures – A Pseudo code Approach with C, 2nd Edition, Cengage Learning
4. Robert Kruse, C L Tondo, Bruce Leung and Shashi Mogalla: Data Structures and Program Design in C, 2nd Edition, Pearson Education

Subject Name: Computer Networks			
Subject Code:	17MCA25	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	04
Detailed Syllabus			
<p>Module -1: 10 Hours Introduction to Computer Networks and Physical Layer Networking Devices, Classification of Computer Networks, Network Protocol Stack (TCP/IP and ISO-OSI), Network Standardization and Examples of Networks. Data Transmission Concepts, Analog and Digital Data Transmission, Transmission Impairments and Channel Capacity, Guided and Wireless transmission, communication media, Digital modulation techniques (FDMA, TDMA, CDMA) and mobile telephone systems (1G, 2G, 3G and 4G).</p> <p>Module -2: 10 Hours Data Link layer Data link layer design issues, Error Detection and Correction Codes, Data Link Protocols and Sliding window protocols.</p> <p>Medium Access Sub Layer The Channel Allocation Problem, Multiple access protocols and Examples: Wireless LAN, Bluetooth.</p> <p>Module -3: 10 Hours Network Layer Network Layer Design issues, Routing algorithms, Congestion Control Algorithms, Quality of Service, Internetworking and The Network Layer in the Internet.</p> <p>Module -4: 10 Hours The Transport Layer The Transport Service, Elements of Transport Protocols, Congestion Control, The Internet Transport Protocol: UDP, The Internet Transport Protocols – TCP, Performance Issues.</p> <p>Module -5: 10 Hours The application Layer DNS: Domain Name Space, Domain Resource Records, Domain Name Servers. Electronic mail: SMTP, The World Wide Web: Static and dynamic web pages, web applications, HTTP, mobile web. Streaming audio and Video: Digital audio and video, streaming stored and line media, real-time conferencing, Content Delivery: content and internet traffic, server forms, web proxies, content delivery networks, peer-to-peer networks.</p> <p>Text Books: 1. "Computer Networks" by Andrew S Tanenbaum, David J Wetheral, 5th Edition, Pearson 2012 (Chapter 1, 2.2, 2.3, 2.5, 2.7, 3.1, 3.2, 3.3, 3.4, 4.1, 4.2, 4.4, 4.6) Chapter 5, Chapter 6 (excluding 6.7), Chapter 7.</p>			

Subject Name: Operating System LAB			
Subject Code:	17MCA26	CIE Marks	10
No of Hours/Week:	4	SEE Marks	40
Total Hours	50	Credits	2
Detailed Syllabus			
<ol style="list-style-type: none"> 1. Write a C program to simulate the following non-preemptive CPU scheduling algorithms to find turnaround time and waiting time. a) FCFS b) SJF c) Round Robin (pre-emptive) d) Priority 2. Write a C program to simulate multi-level queue scheduling algorithm considering the following scenario. All the processes in the system are divided into two categories – system processes and user processes. System processes are to be given higher priority than user processes. Use FCFS scheduling for the processes in each queue 3. Write a C program to simulate the following file allocation strategies. a) Sequential b) Indexed c) Linked 4. Write a C program to simulate the following contiguous memory allocation techniques a) Worst-fit b) Best-fit c) First-fit 5. Write a C program to simulate paging technique of memory management 6. Write a C program to simulate Bankers algorithm for the purpose of deadlock avoidance. 7. Write a C program to simulate disk scheduling algorithms a) FCFS b) SCAN c) C-SCAN 8. Write a C program to simulate page replacement algorithms a) FIFO b) LRU c) LFU <p>Note: In exam student should execute any one of the above program unix/linux OS</p>			

Subject Name: Data Structures Lab Using C++			
Subject Code:	17MCA27	CIE Marks	10
No of Hours/Week:	4	SEE Marks	40
Total Hours	50	Credits	2
Detailed Syllabus			
Laboratory Experiments:			
<ol style="list-style-type: none"> Write a program to convert a prefix notation to postfix notation. Write a program to Evaluate a given postfix expression and its values for the variables Write a program to Simulate the working of circular queue providing the following operations–Insert, Delete and Display. Write a program to Demonstrate recursion ,Generate Fibonacci sequence,Solve Towers of Hanoi Problem. Write a program to Simulate the working of a linked list providing the following operations: Insert at the beginning/end; Insert at the position; Display list Write a program to Simulate the working of a circular linked list providing the following operations Delete from the beginning/end; Delete a given element; Display list Write a program to Simulate the working of a dequeue Write a program to Simulate the working of a double linked list to implement stack. Write a program to Create a binary tree and implement the tree traversal techniques of inorder, preorder and postorder. Write a program to Implement quick sort Write a program to Implement the search techniques of Linear Search; Binary Search Write a program to create a class called STACK using an array of integers. Implement the following operations by overloading the operators '+' and '--': $s1 = s1 + \text{element}$; where s1 is an object of the class STACK and element is an integer to be pushed on the top of the stack $s1 = --s1$; where s1 is an object of the class STACK. '--' operator pops the element. Handle the STACK empty and full conditions. Also display the contents of the stack after each operation, by overloading the << operator. Write a program to create a class called QUEUE with member functions to add an element and to delete an element from the queue. Using the member functions, implement a queue of integers. Demonstrate the operations by displaying the contents of the queue after every operation. 			
Text Book:			
<ol style="list-style-type: none"> H. Schild: C++ The Complete Reference, 4th Edition, Tata McGraw Hill, 2007. 			
Reference Books:			
<ol style="list-style-type: none"> Stanley B.Lippmann, Josee Lajore: C++Primer, 4th Edition, Addison Wesley, 2005. Venugopal K R ,Rajkumar,.Ravishankar T: Mastering C++, Tata McGraw Hill,2007. 			

Subject Name: Database Management Systems Laboratory			
Subject Code:	17MCA28	CIE Marks	10
No of Hours/Week:	4	SEE Marks	40
Total Hours	50	Credits	02
Detailed Syllabus			
<p><i>DBMS Lab Experiments:</i></p> <p>Instructions for the Exercises:</p> <ol style="list-style-type: none"> 1. Draw ER diagram based on given scenario with various Constraints. 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 <p>1 Create the following tables with properly specifying Primary keys, Foreign keys and solve the following queries.</p> <p>BRANCH(Branchid,Branchname,HOD) STUDENT(USN,Name,Address,Branchid,sem) BOOK(Bookid,Bookname,Authorid,Publisher,Branchid) AUTHOR(Authorid,Authurname,Country,age) BORROW(USN,Bookid,Borrowed_Date)</p> <p>) Queries:</p> <ol style="list-style-type: none"> 1 List the details of Students who are all Studying in 2nd sem MCA. 2 List the students who are not borrowed any books. 3 Display the USN, Student name, Branch_name, Book_name, Author_name, Books_Borrowed_Date of 2nd sem MCA Students who borrowed books. 4 Display the number of books written by each Author. 5 Display the student details who borrowed more than two books. 6 Display the student details who borrowed books of more than one Author. 7 Display the Book names in descending order of their names. 8 List the details of students who borrowed the books which are all published by the same Publisher. <p>2 Design an ER-diagram for the following scenario, Convert the same into a relational model and then solve the following queries.</p> <p>Consider a Cricket Tournament "ABC CUP" organized by an organization. In the tournament there are many teams are contesting each having a Teamid,Team_Name, City, a coach. Each team is uniquely identified by using Teamid. A team can have many Players and a captain. Each player is uniquely identified by Playerid, having a Name, and multiple phone numbers,age. A player represents only one team. There are many Stadiums to conduct matches. Each stadium is identified using Stadiumid, having a stadium_name,Address (involves city,area_name,pincode).A team can play many matches. Each match played between the two teams in the scheduled date and time in</p>			

to record in the database. For each match man_of_the match award given to a player.

Queries:

- 1 Display the youngest player (in terms of age) Name, Team name , age in which he belongs of the tournament.
- 2 List the details of the stadium where the maximum number of matches were played.
- 3 List the details of the player who is not a captain but got the man_of _match award at least in two matches.
- 4 Display the Team details who won the maximum matches.
- 5 Display the team name where all its won matches played in the same stadium.

3 Consider the following Scenario and design an ER-Diagram, map the designed ER-diagram into a Relational model.

Consider an organization "ABC" having many employees. An employee works for one department. Each employee identified by using Empid, having Name, address (described as House_no, city, district, state, pin code) and more than one phone numbers. Department identified by using Dno, having Dname, Dlocation. Each Department having a manager . Each department having many employees. There are many Projects , each project is controlled by the department. Each Project uniquely identified by Pno, having Project_name,Project_location. An employee works on many Projects. Number of hours per week worked on each project by an Employee also needs to be recorded in the database . A project is worked by many employees. Each employee supervised by the supervisor. Employee having many dependents. Dependents having the dependent_name, gender, age, address. Dependents are identified by Empid.

T1(Empid, Emp_Name,city, district, state, pin_code, phoneno, Dno,Dname,Dlocation, Dept_mgr_id, Pno, Project_name, Project_location, Number_of_Hours,Supervisor_Empid, Dependent_name, gender, address)

Deduce the above Relation T1 into the 3NF and then solve the following queries. Queries:

1. Display the details of the employees who are working on both the projects having project_no 5 and 10.
2. Display the details of employees having atleast two dependents.
3. Display the project name on which more number of employees are working.
4. Retrieve the employees who do not have any dependents.
5. Display the Employee details whose total number of hours per week working on various projects is maximum than all other employees.
6. create a view to display the number of employees working in each department.

4 Design an ER-diagram for the following scenario, Convert the same into a relational model, normalize Relations into a suitable Normal form and then solve the following queries.

A country can have many Tourist places . Each Tourist place is identified by using tourist_place_id, having a name, belongs to a state, Number of kilometers away from the capital city of that state,history. There are many Tourists visits tourist places every year. Each tourist is identified uniquely by using Tourist_id, having a Name, age, Country and multiple emailids. A tourist visits many Tourist places, it is also required to record the

visted_date in the database. A tourist can visit a Tourist place many times at different dates. A Tourist place can be visited by many tourists either in the same date or at different dates.

Queries:

- 1 List the state name which is having maximum number of tourist places.
- 2 List details of Tourist place where maximum number of tourists visited.
- 3 List the details of tourists visited all tourist places of the state "KARNATAKA".
- 4 Display the details of the tourists visited at least one tourist place of the state, but visited all states tourist places.
- 5 Display the details of the tourist place visited by the tourists of all country.

5 Design an ER-diagram for the following scenario, Convert the same into a relational model, normalize Relations into a suitable Normal form and then solve the following queries.

A country wants to conduct an election for the parliament. A country having many constituencies. Each constituency is identified uniquely by Constituency_id, having the Name, belongs to a state, Number_of_voters. A constituency can have many voters. Each voter is uniquely identified by using Voter_id, having the Name, age, address (involves Houseno,city,state,pincode). Each voter belongs to only one constituency. There are many candidates contesting in the election. Each candidates are uniquely identified by using candidate_id, having Name, phone_no, age, state. A candidate belongs to only one party. There are many parties. Each party is uniquely identified by using Party_id, having Party_Name, Party_symbol. A candidate can contest from many constituencies under a same party. A party can have many candidates contesting from different constituencies. No constituency having the candidates from the same party. A constituency can have many contesting candidates belongs to different parties. Each voter votes only one candidate of his/her constituency.

Queries:

- 1 List the details of the candidates who are contesting from more than one constituencies which are belongs to different states.
- 2 Display the state name having maximum number of constituencies.
- 3 Create a stored procedure to insert the tuple into the voter table by checking the voter age. If voter's age is at least 18 years old, then insert the tuple into the voter else display the "Not an eligible voter msg" .
- 4 Create a stored procedure to display the number_of_voters in the specified constituency Where the constituency name is passed as an argument to the stored procedure.
- 5 Create a TRIGGER to UPDATE the count of " Number_of_voters" of the respective constituency in "CONSTITUENCY" table , AFTER inserting a tuple into the "VOTERS" table.

Note 1: In the practical Examination each student has to pick one question from a lot of all the 5 questions.

Note2: Change of program is not permitted in the Practical Examination

SEMESTER-III

Subject Name: Programming Using Java			
Subject Code:	17MCA31	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	4
Detailed Syllabus			
<p>Module – 1: 8 Hours Fundamentals of Java Language Fundamentals of Java Language – Introduction to Java, Genesis of Java, Why Java?, History of Java, Oak, Java Features, Difference between Java and C++, The Java Virtual Machine, Java Program Structure, Java Tokens, Data Types in Java, Arrays, Operators, Control Statements – Selection Statement, Iteration Statement, Switch Statement</p> <p>Creating Classes and Methods Defining Classes, Creating instance and Classes Variables, Defining instance Variables, Creating Objects, this Keyword, Passing Arguments to methods, Accessing Class Members, Constructors, Default (implicit), Explicit Constructor, Parameterized Constructor, Constructor Overloading, Java.lang.Object, toString(), overriding equals() method, hashCode(), Method Overloading, static members, Nesting of Methods</p> <p>Module – 2: 10 Hours Inheritance: Extending a Class, Defining a subclass, Subclass Constructor, Multilevel Inheritance, Hierarchical Inheritance, super keyword, Visibility Control, public access, private access, protected Access, Creating Java Applications, Overriding Methods, final variables and methods, final classes, finalizer method, Dynamic Method Dispatch, Java Applications and Command Line arguments, Abstract methods and Abstract Classes, Inner Classes, Access Control for the methods and variables</p> <p>Interfaces Defining and implementing an Interface, implements keyword, Achieving Polymorphism by using Interface references, implementing an interface partially, Implementing Multiple Inheritance concept through interfaces, Differences between the Abstract Class and Interfaces, Variables in Interfaces, Extending Interfaces</p> <p>Module – 3: 11 Hours Packages and Wrapper Classes Java API Packages, The import Statement, Using System Packages, Naming Conventions, CLASSPATH Settings, Creating and using Packages, Importing from other Packages, Access protection in Packages, Example for package. Introduction to Wrapper Classes: The Integer class, The Long class, The Float class, The Double</p>			

class, The Character class

Enumerations, Autoboxing and Annotations (Metadata)

Enumerations: Enumeration Fundamentals, the values() and valueOf() Methods, Example.
Autoboxing: Autoboxing and Methods, Autoboxing, Autoboxing / Unboxing Boolean and Character values
Annotations (Metadata): Annotation Basics, Specifying a Retention Policy, Obtaining Annotations at Run Time by Use of Reflection.

String and String Buffer

Introduction to Strings, String Constructors, String Length, String Operations, Additional String Methods, String Buffer, String Builder

Module – 4: 10 Hours

Java.util: The Collections Framework

Introduction to Collections Framework, Collections Overview, The Collection Interfaces, The List Interface, The ArrayList Class, The LinkedList Class, The Vector Class, The Queue Interface, Set Interface , The HashSet, LinketHashSet, The TreeSet Class, MapInterface, HashMap, LinkedHashMap, TreeMap, IteratorInterface, Comparator, Comparable

Exception Handling

Exceptions, Managing Exceptions, Syntax of Exception handling, Java Keywords for handling Exception, try, catch, throw, throws, finally, Multiple Catch Statements, Using finally Block (Statement), Creating a user Defined Exception, Understanding the usage of throws, Declaring methods that might throw exceptions, Creating and throwing your own Exceptions.

Module – 5: 11 Hours

Java I/O Streams

File: Directories, Using FileNameFilter, The listFiles, Creating Dirctrories, The Stream Classes: Byte Stream and Character Stream, Byte Streams: Input Stream Classes, Output Stream Classes
Character Streams: Reader Classes, Writer Classes, Creating Byte Streams using FileInputStream / FileOutputStream Classes, Creating Character Streams using FileReader / FileWriter Classes, Accepting the input from the keyboard using InputSreamReader, BufferedReader and System.in, Creating DataInputStream / DataOutputStream, Serialization

Multi Threading

Multi Tasking & Processes, What is Java Threads?, Definition Multithreading , Multi Threading in Java, Thread Creation Diagram, The Life Cycle of a Thread, Thread Creation: Extending Thread Class and Implementing Runnable Interface, Thread Priorities, Thread Synchronization, Producer – Consumer Problem and wait() and notify() Methods, Suspending and Resuming Threads

Text Book:

1. Herbert Schildt: The Complete Reference JAVA, 7th Edition, Tata McGraw Hill, 2006.

Reference Books:

1. Y. Daniel Liang: Introduction to JAVA Programming, 6th Edition, Pearson Education, 2007.

Subject Name: Analysis and Design of Algorithms			
Subject Code:	17MCA32	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	4
Detailed Syllabus			
<p>Module -1: 10 Hours Notion of Algorithm, Fundamentals of Algorithmic Problem Solving, Important Problem Types, Basics of data Structures. Analysis Framework, Asymptotic Notations and Basic efficiency classes. Mathematical analysis of Recursive and Non-recursive algorithms, Examples. Algorithm visualization. Brute Force Method: Selection Sort and Bubble Sort, Sequential Search and String Matching, Exhaustive Search.</p> <p>Module -2: 13 Hours Divide-and-Conquer: Mergesort, Quicksort, Binary Search, Binary tree Traversals and related properties, Multiplication of large integers, Strassen's Matrix Multiplication. Decrease-and-Conquer: Insertion Sort, Depth First and Breadth First Search, Topological sorting, Algorithms for Generating Combinatorial Objects.</p> <p>Module -3: 10 Hours Transform-and-Conquer: Presorting, Balanced Search Trees. Space and Time Tradeoffs: Sorting by Counting, Input Enhancement in String Matching, Hashing, B-Trees</p> <p>Module -4: 10 Hours Dynamic Programming: Computing a binomial coefficient, Warshall's and Floyd's Algorithms Greedy Technique: Prim's Algorithm, Kruskal's Algorithm.</p> <p>Module -5: 7 Hours Dijkstra's Algorithm, Huffman Trees Limitations of algorithm power, Lower bound arguments, decision trees, P, NP and NP complete problems</p> <p>Text Book:</p> <ol style="list-style-type: none"> Anany Levitin: Introduction to the Design and Analysis of Algorithms, 3rd Edition, Pearson Education. (Chapters 1.1-1.4, 2.1-2.5, 2.7, 3.1, 3.2, 3.4, 4.1-4.5, 5.1-5.4, 6.1, 6.3, 6.4, 6.6, 7.1-7.4, 8.1- 8.3, 9.1-9.4, 11.1-11.3) <p>Reference Books:</p> <ol style="list-style-type: none"> Horowitz E., Sahani S., Rajasekharan S.: Fundamentals of Computer Algorithms, 2nd Edition, Universities Press, 2007. Coremen T.H., Leiserson C.E., and Rivest R.L.: Introduction to Algorithms, 2nd Edition, PHI, 2001 			

Subject Name: Software Engineering			
Subject Code:	17MCA33	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	4
Detailed Syllabus			
<p>Module -1: 10 Hrs Overview Introduction: Professional Software Development Attributes of good software, software engineering diversity, IEEE/ ACM code of software engineering ethics, case studies</p> <p>Software Process & Agile Software Development Software Process models: waterfall, incremental development, reuses oriented, Process activities; Coping with change, The rational Unified process. Agile methods, Plan-driven and agile Development, Extreme Programming, Agile project management, Scaling agile methods.</p> <p>Module -2: 10 Hours Requirements Engineering Functional and non-functional requirements, The software requirements document, Requirements specification, Requirements engineering processes, Requirement elicitation and analysis, Requirements validation, Requirements management</p> <p>Component-based software engineering Components and component model, CBSE process, Component composition.</p> <p>Module -3: 10 Hours System Modeling, Architectural Design Context models, Interaction models, Structural models, Behavioral models, Model-driven engineering, Software architecture: the role of software architecture, architectural views, component and connector view, Architectural styles for C&C view, Documenting architectural design</p> <p>Module -4; 10 Hours Design and implementation Design: Design concepts, Function oriented design, detailed design, verification, matrix (Complexity matrix for function oriented design)</p> <p>Distributed Software engineering Distributed system issues, Client-server computing, Architectural patterns for distributed systems, Software as a service.</p> <p>Module -5: 10 Hours Planning a software Project</p>			

management plan, Quality plan, Risk Management, Project monitoring plan.

Software Testing

Testing fundamentals, Black-box testing, White-box testing, Testing process

Text Books:

1. Ian Sommerville : Software Engineering, 9th edition, Person Education Ltd, 2011.
(Chapters:- 1, 2, 3, 4, 5, 17, 18)
2. Pankaj Jalote: 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.

Subject Name: Python Programming			
Subject Code:	17MCA34	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	4
Detailed Syllabus			
<p>Module -1: 10 Hrs Installing Python, Simple program using Python, Expressions and Values, Variables and Computer Memory, error detection, Multiple line statements, Designing and using functions, functions provided by Python, Tracing function calls in memory model, omitting return statement. Working with Text: Creating Strings of Characters, Using Special Characters in Strings, Creating a Multiline String, Printing Information, Getting Information from the Keyboard.</p> <p>Module -2; 10 Hrs A Boolean Type , Choosing Statements to Execute, Nested If Statements , Remembering the Results of a Boolean Expression Evaluation , A Modular Approach to Program Organization, Importing Modules , Defining Your Own Modules, Testing Code Semi automatically Grouping Functions Using Methods: Modules, Classes, and Methods , Calling Methods the Object- Oriented Way, Exploring String Methods, Underscores.</p> <p>Module -3; 10 Hrs Storing Collections of Data Using Lists: Storing and Accessing Data in Lists, modifying Lists, Operations on Lists, Slicing Lists, Aliasing, List Methods, Working with a List of Lists. Repeating Code Using Loops: Processing Items in a List, Processing Characters in Strings, Looping Over a Range of Numbers, Processing Lists Using Indices, Nesting Loops in Loops, Looping Until a Condition Is Reached, Repetition Based on User Input, Controlling Loops Using Break and Continue Reading and Writing</p> <p>Module -4: 10 Hrs Files: Kinds of files, Opening a File, Techniques for Reading Files, Files over the Internet, Writing Files, and Writing Algorithms That Use the File-Reading Techniques, Multiline Records. Storing Data Using Other Collection Types: Storing Data Using Sets, Storing Data Using Tuples, Storing Data Using Dictionaries, Inverting a Dictionary, Using the In Operator on Tuples, Sets, and Dictionaries, Comparing Collections.</p> <p>Module -5: 10 Hours Collection of New Information Object-Oriented Programming : Understanding a Problem Domain , Function "Is instance," Class Object, and Class Book , Writing a Method in Class Book, Plugging into Python Syntax: More Special Methods ,Creating Graphical User interface: Building a Basic GUI, Models, Views, and Controllers, Customizing the Visual Style Introducing few more Widgets, Object-Oriented GUIs, Keeping the Concepts from Being a GUI Mess.</p>			

Question paper pattern:

- The question paper will have ten questions.
- Each full question consists of 16 marks.
- There will be 2 full questions (with a maximum of four sub questions) from each module.
- Each full question will have sub questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Practical Programming: An introduction to Computer Science Using Python, second edition, Paul Gries, Jennifer Campbell, Jason Montojo, The Pragmatic Bookshelf.
2. Learning with Python: How to Think Like a Computer Scientist Paperback – Allen Downey , Jeffrey Elkner, 2015

Reference Books:

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

Subject Name: Computer Graphics			
Subject Code:	17MCA35	CIE Marks	20
No of Hours/Week:	04	SEE Marks	80
Total Hours	50	Credits	04
Detailed Syllabus			
<p>Module1: 10 Hours Graphics Output Primitives and Attributes Introduction to open GL, Coordinate reference frames, Specifying two dimensional world coordinate reference frame in Open GL, Open GL point functions, Open GL line functions, Line drawing algorithms, Circle generation algorithms, Ellipse generation algorithms, Fill area primitives, Polygon fill areas, OpenGL polygon fill area functions, General scan line polygon fill algorithm, Fill methods for areas with irregular boundaries, Open GL fill area attribute functions</p> <p>Module 2:12 Hours Two – Dimensional and Three - Dimensional Geometric Transformations Basic two dimensional geometric transformations, Matrix representations and homogeneous coordinates, Inverse transformations, Two dimensional composite transformations, Other two dimensional transformations, Three dimensional Translation, Rotation, Scaling, Other three dimensional transformations, Affine transformations, Open GL geometric transformation functions</p> <p>Module 3: 10 Hours Two Dimensional Viewing The two dimensional viewing, Clipping window, Normalization and viewport transformations, Clipping algorithms, Two dimensional point clipping, Two dimensional line clipping algorithms, Polygon fill area clipping, Curve clipping, Text clipping</p> <p>Module 4: 10 Hours Three Dimensional Viewing The three dimensional viewing concepts, Three dimensional viewing pipeline, Three dimensional viewing coordinate parameters, Transformation from world to viewing coordinates, Projection transformations, Orthogonal projections, Oblique parallel projections, Perspective projections, The viewport transformation and three dimensional screen coordinates</p> <p>Module 5: 08 Hours Curves and Computer Animation Bezier spline curves, Raster methods for computer animation, Design of animation sequences, Traditional animation techniques, General computer animation functions</p> <p>Text book: Donald Hearn, M.Pauline Baker, Computer Graphics with Open GL, Pearson (Indian</p>			

Reference Books:

1. Edward Angel, „Interactive Computer Graphics“ – A top down approach using Open GL, Pearson, Fifth Edition
2. Peter Shirley, Steve Marschner, 'Computer Graphics, Cengage Learning (Indian edition), 2009.

Subject Name: Java Programming Laboratory			
Subject Code:	17MCA36	CIE Marks	10
No of Hours/Week:	4	SEE Marks	40
Total Hours	50	Credits	02
Detailed Syllabus			
<ol style="list-style-type: none"> 1. Programs on classes and objects 2. Programs on Inheritance 3. Programs on abstract class and inner class 4. Programs on interfaces 5. programs on packages 6. programs on string and string buffer classes 7. Programs on Collections 8. programs on exception handling 9. programs on IO streams 10. programs on Multithreading 			

Subject Name: Analysis & Design of Algorithms Laboratory			
Subject Code:	17MCA37	CIE Marks	10
No of Hours/Week:	4	SEE Marks	40
Total Hours	50	Credits	02
Detailed Syllabus			
<ol style="list-style-type: none"> 1. To sort a given set of elements using the Heap Sort method and determine the time required to sort the elements 2. To sort a given set of elements using the Merge Sort method and determine the time required to sort the elements. 3. To sort a given set of elements using Quick sort method and determine the time required sort the elements. 4. From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm <ol style="list-style-type: none"> a) To implement Floyd's algorithm for the All-Pair-Shortest-Path problem. b) To find the binomial coefficient using dynamic programming. 5. To implement 0/1 Knapsack problem using dynamic programming. 6. To print all the nodes reachable from a given starting node in a digraph using BFS method. 7. To check whether a given graph is connected or not using DFS method. 8. To implement Horspool algorithm for string matching. 9. To compute the transitive closure of a given directed graph using Warhsall's algorithm. To implement N queen's problem using Back Tracking 			

Subject Name: Python Programming Laboratory			
Subject Code:	17MCA38	CIE Marks	10
No of Hours/Week:	4	SEE Marks	40
Total Hours	50	Credits	02
Detailed Syllabus			
<p>Laboratory Experiments: NOTE: The experiments are to be carried using discrete components only.</p> <ol style="list-style-type: none"> 1. Write a program to sum all the elements from n1 to n2 where n1 and n2 are positive integers 2. Input an array of n numbers and find separately the sum of positive numbers and negative numbers. 3. Write a program to search an element using linear search 4. Write a program to search an element using binary search. 5. Write a program to simulate stack. 6. Using a stack evaluate an arithmetic expression. 7. Write a program to multiply two matrices. 8. Write a program to find the roots of a quadratic equation 9. Insert a number in a sorted array. 10. Write a Python Program to check whether the given string is palindrome or not using built in string manipulation methods. 11. Write a Python Program to read a word and prints the number of letters, vowels and percentage of vowels in the word using dictionary 12. Write a Python Program to check a given sentence is a pangram or not using function/Module. 13. Write a Python Event driven Program for file operations Press 1: to open file in read mode 2: open the file in write mode 3: current position of the file pointer #4: Reposition the pointer at the beginning 5: exit. 14. Write an Object oriented Python program to create two Time objects: currentTime, which contains the current time; and breadTime, which contains the amount of time it takes for a bread maker to make bread. Then we'll use addTime to figure out when the bread will be done. Write the printTime function to display the time when the bread will be done by the bread 			

SEMESTER-IV

Subject Name: Advanced Java Programming			
Subject Code:	17MCA41	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	4
Detailed Syllabus			
<p>Module -1; 10 Hrs</p> <p>Servlet</p> <p>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.</p> <p>Module -2: 12 Hours</p> <p>JSP and Controlling the Structure of generated servlets</p> <p>Overview of 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, import attribute, session attribute, isElignore attribute, buffer and autoflush attributes, info attribute, errorPage and isErrorPage attributes, isThreadSafe Attribute, extends attribute, language attribute, Including files and applets in jsp Pages, using java beans components in JSP documents</p> <p>Module – 3: 7 Hours</p> <p>Annotations and Java Beans</p> <p>Creating Packages, Interfaces, JAR files and Annotations. The core java API package, New java.Lang Sub package, Built-in Annotations with examples. Working with Java Beans. Introspection, Customizers, creating java bean, manifest file, Bean Jar file, new bean, adding controls, Bean properties, Simple properties, Design Pattern events, creating bound properties, Bean Methods, Bean an Icon, Bean info class, Persistence, Java Beans API.</p> <p>Module – 4: 09 Hours</p> <p>Talking to Database, Immediate Solutions, Essential JDBC program, using prepared</p>			

Statement Object, Interactive SQL tool. JDBC in Action Result sets, Batch updates, Mapping, Basic JDBC data types, Advanced JDBC data types, immediate solutions.

Module -5; 12 Hours

Introduction to EJB and Server Side Component Models

The Problem domain, Breakup responsibilities, Code Smart not hard, the Enterprise java bean specification. Components Types. Server Side Component Types, Session Beans, Message Driven Beans, Entity Beans, The Java Persistence Model. Container services. Dependency Injection, Concurrency, Instance pooling n caching, Transactions, security, Timers, Naming and object stores, Interoperability, Life Cycle Callbacks, Interceptors, platform integration. Developing your first EJB. Preparation, Definitions, naming conventions, convention for the Examples, coding the EJB, the contract, the bean Implementation class, out of Container Testing, Integration Testing. Models: The Stateless Session Bean, the Stateful Session Bean, the Singleton Session Bean, Message-Driven Beans. EJB and PERSISTENCE. Persistence Entity manager Mapping Persistence objects, Entity Relationships

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
3. Andrew Lee Rubinger, 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

Subject Name: Advanced Web Programming			
Subject Code:	17MCA42	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	4
Detailed Syllabus			
<p>Module -1: 10 Hrs Programming in Perl and CGI Scripting and Building Web Applications with Perl Origins and uses of Perl, Scalars and their operations, Assignment statements and simple input and output, Control statements, Fundamentals of arrays, Hashes, References, Fun, What is CGI? Developing CGI Applications actions, Pattern matching, File input and output; Examples. CGI.pm methods, Example, Creating HTML Pages Dynamically, Using CGI. pm An Example, Adding Robustness, libwww, Carp, Cookies, Uploading files, Tracking users with Hidden Data, Using Relational Databases</p> <p>Module -2: 10 Hours Introduction to PHP and Building Web applications with PHP Origins and uses of PHP, Overview of PHP, General syntactic characteristics, Primitives, operations and expressions, Output, Control statements, Arrays, Functions, Pattern matching, Form handling, Files, Tracking users, cookies, sessions, Using databases, Handling XML</p> <p>Module -3: 10 Hours Introduction to Ruby and Introduction to Rails Origins and uses of Ruby, Scalar types and their operations, Simple input and output, Control statements, Arrays, Hashes, Methods, Classes, Code blocks and iterates, Pattern matching. Overview of Rails, Document requests, Processing forms, Layouts. Rails applications with Databases.</p> <p>Module – 4: 10 Hours Introduction to web 2.0 and Web Services What is Web 2.0?, Folksonomies and Web 2.0, Software As a Service (SaaS), Data and Web 2.0, Convergence, Iterative development, Rich User experience, Multiple Delivery Channels, Social Networking. Web Services: SOAP, RPC Style SOAP, Document style SOAP. WSDL, REST services, JSON format, What is JSON? Array literals, Object literals, Mixing literals, JSON Syntax, JSON Encoding and Decoding, JSON versus XML</p> <p>Module -5: 10 Hrs Data Driven Documents: Data visualization tool for web apps</p>			

Introduction to D3: Building a Simple Subway Train Status Board, Graphing Mean Daily Plaza Traffic. Scales, Axes, and Lines, Graphing Turnstile Traffic, Interaction and Transitions, Subway Connectivity, Scheduled Wait Time Distribution.

Text Books:

1. Robert W. Sebesta: Programming the Worldwide Web, 4th Edition, Pearson Education, 2012
Francis Shanahan: Mashups, WileyIndia, 2012
2. Mike Dewar: "Getting Started with D3": O'Reilly Media, 2012

Reference Books:

1. M. Deitel, P.J. Deitel, A.B. Goldberg: Internet & World Wide Web How to program, 3rd Edition, Pearson Education/PHI, 2004

Subject Name: Mobile Applications			
Subject Code:	17MCA43	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	4
Detailed Syllabus			
<p>Module -1: 10 Hrs</p> <p>Introduction</p> <p>Preliminary Considerations – Cost of Development – Importance of Mobile Strategies in the Business World – Effective use of Screen Real Estate –</p> <p>Understanding Mobile Applications</p> <p>Understanding Mobile Applications Users – Understanding Mobile Information Design – Understanding Mobile Platforms – Using the Tools of Mobile Interface Design.</p> <p>Module -2: 10 Hours</p> <p>Getting Started with Android Programming</p> <p>What is Android – Obtaining the required tools– Anatomy of an Android Application – Components of Android Applications – Activities – Fragments – Utilizing the Action Bar</p> <p>Module -3: 10 Hours</p> <p>Android UI Design and Location Based Services</p> <p>Views and View Groups – Basic Views – Fragments – Displaying Maps – Getting Location Data – Publishing for Publishing – Deploying APK Files</p> <p>Module -4: 10 Hours</p> <p>Android Messaging and Networking</p> <p>SMS Messaging – Sending Email – Networking – Downloading Binary Data, Text files – Accessing Web Services – Performing Asynchronous Call – Creating your own services – Communicating between a service and an activity – Binding activities to services</p> <p>Module -5: 10 Hours</p> <p>Feedback and Oscillator Circuits</p> <p>iOS – Obtaining the tools and SDK – Components of XCODE – Architecture of iOS – Building Derby App in iOS – Other useful iOS things – Windows Phone: Getting the tools you need – Windows Phone 7 Project – Building Derby App in Windows Phone 7 – Distribution – Other useful Windows Phone Thing</p> <p>Text Books:</p> <p>Jeff Mc Wherter and Scott Gowell, “Professional Mobile Application Development”, 1st Edition, 2012, ISBN: 978-1-118-20390-3</p>			

Reference Books:

1. Wei-Meng Lee, “Beginning Android Application Development”, Wiley 2011.
2. Reto Meier, “Professional Android 4 Application Development”, Wrox Publications 2012

Elective-I

Subject Name: Basics of MIS and E-Commerce			
Subject Code:	17MCA441	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	4
Detailed Syllabus			
<p>Module-1: 8 Hours Information Concepts Information concepts, Classification of Information, Methods of Data and Information collection, Value of Information, Information: A quality product, General model of a Human as Information processor.</p> <p>Introduction of MIS MIS: Concept, Definition, Role of the MIS, Impact of MIS, MIS and the user, Management as a control system, MIS support to the management, Management effectiveness and MIS, Organization as system. MIS: organization effectiveness</p> <p>Module-2: 11Hrs Decision Making and DSS Decision making concepts; Decision making process, Decision-making by analytical modeling, Behavioral concepts in Decision making, Organizational Decision-making, Decision structure, DSS components, Management reporting alternatives, Group decision-support systems (GDSS) What is a GDSS and its benefits.</p> <p>Electronic Business systems Enterprise business system –Introduction, Sales force automation, online accounting system, Customer relationship management, Supply chain management.</p> <p>Module-3: 11Hrs Enterprise Resource Planning (ERP) Introduction to ERP, Common myths about ERP, History of ERP, Advantages and disadvantages of ERP, Why is ERP a business necessity? , Basic concepts of ERP, Components of ERP, Suppliers of ERP, Business Process Reengineering BPR and ERP, Selection of ERP, Implementation of ERP, Benefits of ERP packages.</p> <p>Module-4: 10 Hrs E-business Technology on Client Server Architecture Client server architecture, implementation strategies, The Internet and World Wide Web, Intranet/Extranet, Electronic, Impact of Web on Strategic management, Web enabled business management, MIS in Web environment.</p> <p>E-Commerce Introduction Introduction to E-commerce, What is E-commerce? ,Electronic commerce-Architectural</p>			

Module-5: 10 Hrs**E-Commerce Applications**

E-Commerce Applications: Business-to-Consumer (B2C), Consumer-to-Consumer (C2C), Business-to-Business (B2B), Market places, and Communities, Web Security, Building an E-Commerce Web Site.

E-Commerce Payment Systems

E-Commerce Payment Systems, Electronic payment systems, Digital Token-Based, Smart Cards, Credit Cards, Risks in Electronic payment systems, EDI: Electronic Data Interchange.

Text Books:

1. Waman S Jhawadekar: Management Information System, 3rd Edition, Tata McGraw Hill
2. James A O'Brien and George M Marakas: Management Information System, 7th Edition, Tata McGraw Hill, 2006
4. Frontiers of Electronic Commerce-Ravi Kalakota, Andrew Winston.
5. H. M. Deitel, P. J. Deitel and T. R. Nieto, E-Business and E-Commerce: How to Program, Prentice hall, 2001
6. Enterprise Resource Planning ALEXIS LEON, 2nd Edition.

Reference Books:

1. Henry Chan, Raymond Lee, Tharam Dillon, Elizabeth Chang: E-Commerce Fundamentals and Applications, Wile India Edition
2. Ralph M Stair and George W Reynolds: Principles of Information Systems, 7th Edition, Thomson, 2010.
3. Steven Alter: Information Systems -The Foundation of E-Business, 4th Edition, Pearson Education, 2001
4. Rahul De, Managing Information Systems in Business, Government and Society, Wiley India.

Subject Name: Enterprise Resource Planning			
Subject Code:	17MCA442	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	4
Detailed Syllabus			
Module -1; 8 Hrs			
Enterprise Resource Planning –Introduction			
What is ERP? Need of ERP, Advantages of ERP, Growth of ERP			
Module -2: 12 Hrs			
Business process Reengineering (BPR), Supply Chain Management (SCM) ,Online Analytical Processing (OLTP).Management Information System (MIS),Decision Support Systems (DSS),Executive Support Systems (ESS),Data Warehousing, Data Mining Supply Chain Management (SCM),Customer Relationship Management (CRM).			
Module -3: 12 Hrs			
ERP modules & Vendors Finance			
Evaluation and selection of ERP package Production planning, control & maintenance Sales & Distribution Human Resource Management (HRM) Inventory Control System, Quality Management, ERP Market, Project planning,Implementation team training & testing,End user training & Going LivePost Evaluation & Maintenance.			
Module -4: 10 Hrs			
ERP Implementation Life Cycles Evaluation and selection of ERP package Project planning Implementation team training & testing End user training & Going Live Post Evaluation & Maintenance (3)			
Module-5: 8 hours ERP			
Post implementation review of ERP Packages in Manufacturing, Services and other Organizations (3)			
Text Books:			
1. Alexis Leon: Enterprise Resource Planning, 2nd edition TMH publication. V.K. Garg & N.K. Venkitakrishnan: ERP Implementation Framework, PHI.			

Subject Name: Introduction to AI			
Subject Code:	17MCA443	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	4
Detailed Syllabus			
<p>Module -1: 10 Hrs What is Artificial Intelligence: The AI Problems, The Underlying assumption, What is an AI Technique?, The Level of the model, Criteria for success, some general references, One final word and beyond. Problems, problem spaces, and search: Defining, the problem as a state space search, Production systems, Problem characteristics, Production system characteristics, Issues in the design of search programs, Additional Problems.</p> <p>Module -2: 10 Hrs Heuristic search techniques: Generate-and-test, Hill climbing, Best-first search, Problem reduction, Constraint satisfaction, Mean-ends analysis. Knowledge representation issues: Representations and mappings, Approaches to knowledge representation, Issues in knowledge representation, The frame problem. Using predicate logic: Representing simple facts in logic, representing instance and ISA relationships, Computable functions and predicates, Resolution, Natural Deduction</p> <p>Module 3: 10 Hours Symbolic Reasoning Under Uncertainty: Introduction to nonmonotonic reasoning, Logic for nonmonotonic reasoning, Implementation Issues, Augmenting a problem-solver. Implementation: Depth-first search, Implementation: Breadth-first search. Statistical Reasoning: Probability and Bayes Theorem, Certainty factors and rule-based systems, Bayesian Networks, Dempster-Shafer Theory, Fuzzy logic.</p> <p>Module -4: 10 Hours Weak Slot-and-filter structures: Semantic Nets Frames, Strong slot-and -filler structures: Conceptual dependency, scripts, CYC</p> <p>Module -5: 10 Hours Advanced Topics in AI: Minimax search, Goal Stack Planning, Expert System-Representation, Expert System Shell, explanation, Knowledge acquisition.</p> <p>Text Books: 1. Elaine Rich, Kevin Knight, Shiva Shankar B Nair: Artificial Intelligence, Tata McGraw Hill 3rd edition. 2013</p> <p>Reference Books: 1. Stuart Russel, Peter Norvig: Artificial Intelligence A Modern Approach, Pearson 3rd edition 2013. 2. Nils J. Nilsson: "Principles of Artificial Intelligence", Elsevier, ISBN-13: 9780934613101</p>			

Elective-II

Subject Name: Principles of User Interface Design			
Subject Code:	17MCA451	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	4
Detailed Syllabus			
<p>Module-1: 8 Hrs Introduction Usability of Interactive Systems: Introduction, Usability Requirements, Usability measures, Usability Motivations, Universal Usability, Goals for our profession Guideline, principles, and Theories: Introduction, Guidelines, principles, Theories, Object-Action Interface Model</p> <p>Module-2: 10 Hrs Development Processes Managing Design Processes: Introduction, Organizational Design to support Usability, The Three pillars of design, Development Methodologies, Ethnographic Observation, Participatory Design, Scenario Development, Social Impact statement for Early Design Review, Legal Issues.</p> <p>Evaluating Interface Designs Introduction, Expert Reviews, Usability Testing and Laboratories, Survey Instruments, Acceptance Tests, Evaluation During Active Use, Controlled Psychologically Oriented Experiments.</p> <p>Software Tools: Introduction, Specification Methods, Interface-Building Tools, Evaluation and Critiquing Tools.</p> <p>Module-3: 8 Hrs Interaction Styles Direct Manipulation and Virtual Environments: Introduction, Examples of Direct Manipulation, 3D Interfaces, Teleoperation, Virtual and Augmented Reality.</p> <p>Menu Selection, Form Fillin, and Dialog Boxes: Introduction, Task-Related Menu Organization, Single Menus, Combinations of Multiple Menus, Content Organization, Fast Movement Through Menus, Data Entry with Menus: Form Fill-in, Dialog Boxes, and Alternatives, Audio Menus and Menus for small Displays.</p> <p>Module-4: 14 Hrs Command and Natural Languages</p>			

Auditory interfaces, Displays-Small and Large, Printers.

Design Issues

Quality of Service: Introduction, Models of Response-Time Impacts, Expectations and Attitudes, User Productivity, Variability in Response Time, Frustrating Experiences.

Balancing Function and Fashion: Introduction, Error Messages, Non anthropomorphic Design, Display Design, Window Design, Color.

Module-5: 10 Hrs

User Manuals, Online Help, and Tutorials

Introduction, Paper versus Online Manuals, Reading from Paper Verses from Displays, Shaping the Content of the Manuals, Online Manuals and Help, Online Tutorials, Demonstrations, and Guides, Online Communities for User Assistance, The Development Process.

Information Search and Visualization:

Introduction, Search in Textual Documents and Database Querying, Multimedia Document Searches, Advanced Filtering and Search Interfaces, Information Visualization

Text Book:

1. Ben Shneiderman: Designing the User Interface, 4rdEdition, Pearson Education, 2009. (Chapters 1 to 9 and 11 to 14)

Reference Books:

1. Alan J Dix et. al.: Human-Computer Interaction, II Edition, Prentice-Hall India, 1998.
2. Eberts: User Interface Design, Prentice-Hall, 1994.
3. Wilber O Galitz: The Essential Guide to User Interface Design - An Introduction to GUI Design, Principles and Techniques, Wiley-Dreamtech India Pvt. Ltd, 1998.

Subject Name: Multimedia Communications			
Subject Code:	17MCA452	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	4
Detailed Syllabus			
<p>Module-1: 10 Hrs Introduction, Media and Data Streams Multimedia Elements; Multimedia Applications; Multimedia Systems Architecture; Evolving Technologies for Multimedia Systems; Defining Objects for Multimedia Systems; Multimedia Data Interface Standards; The need for Data Compression; Multimedia Databases.</p> <p>Media: Perception Media, Representation Media, Presentation Media, Storage Media, Transmission Media, Information Exchange Media, Presentation Spaces & Values, and Presentation Dimensions; Key Properties of a Multimedia System: Discrete & Continuous Media, Independence Media, Computer Controlled Systems, Integration; Characterizing Data Streams: Asynchronous Transmission Mode, Synchronous Transmission Mode, Isochronous Transmission Mode; Characterizing Continuous Media Data Streams.</p> <p>Module 2: 10 Hrs Audio Technology Sound: Frequency, Amplitude, Sound Perception and Psychoacoustics; Audio Representation on Computers; Three Dimensional Sound Projection; Music and MIDI Standards; Speech Signals; Speech Output; Speech Input; Speech Transmission.</p> <p>Graphics and Images, Video Technology, Computer-Based Animation Capturing Graphics and Images Computer Assisted Graphics and Image Processing; Reconstructing Images; Graphics and Image Output Options. Basics; Television Systems; Digitalization of Video Signals; Digital Television; Basic Concepts; Specification of Animations; Methods of Controlling Animation; Display of Animation; Transmission of Animation; Virtual Reality Modeling Language.</p> <p>Module 3:10 Hrs Data Compression Storage Space; Coding Requirements; Source, Entropy, and Hybrid Coding; Basic Compression Techniques; JPEG: Image Preparation, Lossy Sequential DCT-based Mode, Expanded Lossy DCT based Mode, Lossless Mode, Hierarchical Mode. H.261(Px64) and H.263: Image Preparation, Coding Algorithms, Data Stream, H.263+ and H.263L; MPEG: Video Encoding, Audio Coding, Data Stream, MPEG-2, MPEG-4, MPEG-7; Fractal Compression.</p>			

Module 4: 10 Hrs**Optical Storage Media**

History of Optical Storage; Basic Technology; Video Discs and Other WORMs; Compact Disc Digital Audio; Compact Disc Read Only Memory; CD-ROM Extended Architecture; Further CD-ROM-Based Developments; Compact Disc Recordable; Compact Disc Magneto-Optical; Compact Disc Read/Write; Digital Versatile Disc.

Module 5: 10 Hrs**Data and File Format Standards**

Rich-Text Format; TIFF File Format; Resource Interchange File Format(RIFF); MIDI File Format; JPEG DIB File Format for Still and Motion Images; AVI Index File Format; MPEG Standards; TWAIN

Multimedia Application Design

Multimedia Application Classes; Types of Multimedia Systems; Virtual Reality Design; Components of Multimedia Systems; Organizing Multimedia Databases; Application Workflow Design Issues; Distributed Application Design Issues.

TextBooks:

1. Ralf Steinmetz, Klara Narstedt: Multimedia Fundamentals: Vol1-Media Coding and Content Processing, 2
2. Prabhat K. Andleigh, Kiran Thakrar: Multimedia Systems Design, PHI,2003. (Chapters 1,3,7)

ReferenceBooks:

1. K. R Rao, Zoran S. Bojkovic and Dragorad A. Milovanovic: Multimedia Communication Systems: Techniques, Standards, and Networks, Pearson Education, 2002.
2. Nalin K Sharad: Multimedia Information Networking, PHI,2002

Subject Name: Internet Of Things (IOT)			
Subject Code:	17MCA453	CIE Marks	20
No of Hours/Week:	4	SEE Marks	80
Total Hours	50	Credits	4
Detailed Syllabus			
<p>Module-1:10 Hrs M2M to IoT Introduction: The Vision-Introduction, From M2M to IoT, M2M towards IoT-the global context, A use case example, Differing Characteristics</p> <p>Module-2:10 Hrs M2M to IoT A Market Perspective– Introduction, Some Definitions, M2M Value Chains, IoT Value Chains, An emerging industrial structure for IoT, The international driven global value chain and global information monopolies. M2M to IoT-An Architectural Overview– Building an architecture, Main design principles and needed capabilities, An IoT architecture outline, standards considerations.</p> <p>Module – 3: 10 Hrs M2M and IoT Technology Fundamentals Devices and gateways, Local and wide area networking, Data management, Business processes in IoT, Everything as a Service(XaaS), M2M and IoT Analytics, Knowledge Management</p> <p>Module -4: 10 Hrs IoT Architecture-State of the Art Introduction, State of the art, Architecture Reference Model- Introduction, Reference Model and architecture, IoT reference Model</p> <p>Module-5: 10 Hrs IoT Reference Architecture Introduction, Functional View, Information View, Deployment and Operational View, Other Relevant architectural views. Real-World Design Constraints- Introduction, Technical Design constraints-hardware is popular again, Data representation and visualization, Interaction and remote control. Industrial Automation- Service-oriented architecture-based device integration, SOCRADES: realizing the enterprise integrated Web of Things, IMC-AESOP: from the Web of Things to the Cloud of Things, Commercial Building Automation- Introduction, Case study: phase one-commercial building automation today, Case study: phase two- commercial building automation in the future.</p>			

Text Books:

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, Academic Press, 2014.

References

1. Vijay Madiseti 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

Subject Name: Advance Java lab			
Subject Code:	17MCA46	CIE Marks	10
No of Hours/Week:	4	SEE Marks	40
Total Hours	50	Credits	2
Detailed Syllabus			
<ol style="list-style-type: none"> 1. 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). 2. 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). 3. Write a JAVA Servlet Program to implement and demonstrate get() and Post methods(Using HTTP ServletClass). 4. Write a JAVA Servlet Program using cookies to remember user preferences. 5. a. Write a JAVA JSP Program to implement verification of a particular user login and display a Welcome page. b. Write a JSP program to demonstrate the import attribute. 6. Write a JAVA JSP Program which uses jsp:include and jsp:forward action to display a Webpage. 7. Write a JAVA JSP Program which uses <jsp:plugin> tag to run a applet. 8. 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. 9. Write a JAVA Program to insert data into Student DATA BASE and retrieve info based on particular queries(For example update, delete, search etc...). 10. Write a JSP program to implement all the attributes of page directive tag. 11. An EJB application that demonstrates Session Bean (with appropriate business logic). 12. An EJB application that demonstrates MDB (with appropriate business logic). 13. An EJB application that demonstrates persistence (with appropriate business logic). <p>Note: In the examination each student should any one of the above questions</p>			

Subject Name: Advanced Web Lab			
Subject Code:	17MCA47	CIE Marks	10
No of Hours/Week:	4	SEE Marks	40
Total Hours	50	Credits	02
Detailed Syllabus			
<p>1. Develop and demonstrate a XHTML file that includes Javascript script to generate first n Fibonacci numbers.</p> <p>2. Develop and demonstrate the usage of inline and external style sheet using CSS</p> <p>3. Develop and demonstrate, using Javascript script, a XHTML document that collects the student register number (the valid format is: A digit from 1 to 4 followed by two upper-case characters followed by two digits followed by two upper-case characters followed by three digits; no embedded spaces allowed) of the user. Event handler must be included for the form element that collects this information to validate the input. Messages in the alert windows must be produced when errors are detected.</p> <p>4. 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.</p> <p>5. Design an XML document to store information about a student in a college affiliated to BU. The information must include USN, Name, 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.</p> <p>6. Write a Perl program to display a digital clock which displays the current time of the server.</p> <p>7. Write a Perl program to insert name and age information entered by the user into a table created using MySQL and to display the current contents of this table.</p> <p>8. 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.</p> <p>9. Write a PHP program to read student data from an XML file and store into the MYSQL database. Retrieve and display.</p> <p>10. Write a Perl program to keep track of the number of visitors visiting the web page and to display this count of visitors, with proper headings.</p> <p>11. Write a CGI-Perl program to use a cookie to remember the day of the last login from a</p>			

user and display it when run.

12. Write a Perl program to display various Server informations like Server Name, ServerSoftware, Server protocol, CGI Revision etc.

13. Create a XHTML form with Name, Address Line 1, Address Line 2, and E-mail text fields. On submitting, store the values in MySQL table. Retrieve and display the data based on Name.

14. Write a Perl program to accept the User Name and display a greeting message randomly chosen from a list of 4 greeting message

Note: In examination Student should execute any of the above program.

Subject Name: Mini Project-1			
Subject Code:	17MCA48	CIE Marks	10
No of Hours/Week:	4	SEE Marks	40
Total Hours	50	Credits	02
Detailed Syllabus			
<p>Develop a web application project using the languages and concepts learnt in the theory with a good look and feel effects. You can use any web technologies and frameworks and databases.</p> <p>Note:</p> <ol style="list-style-type: none"> 1. 2. A team of two or three students must develop the mini project. However during the examination, each student must demonstrate the project individually. 3. The team must submit a brief project report 4. The report must be evaluated for 40 Marks. Demonstration and Viva for 40 Marks. 			

V SEMESTER

Subject Name: Object-Oriented Modeling and Design Patterns			
Subject Code	17MCA51	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	52	SEE Hours	03
CREDITS – 04			
Module-1: 10 Hrs			
.			
<p>What is Object Orientation? What is OO development? OO themes; Evidence for usefulness of OO development; OO modeling history. Modeling as Design Technique: Modeling; abstraction; The three models. Class Modeling: Object and class concepts; Link and associations concepts; Generalization and inheritance; A sample class model; Navigation of class models; Practical tips. Advanced object and class concepts; Association ends; N-ary associations; Aggregation; Abstract classes; Multiple inheritance; Metadata; Reification; Constraints; Derived data; Packages; Practical tips. State Modeling: Events, States, Transitions and Conditions; State diagrams; State diagram behavior; Practical tips.</p>			
Module-2: 10 Hrs			
<p>Advanced State Modeling: Nested state diagrams; Nested states; Signal generalization; Concurrency; A sample state model; Relation of class and state models; Practical tips. Interaction Modeling: Use case models; Sequence models; Activity models. Use case relationships; Procedural sequence models; Special constructs for activity models. Process Overview: Development stages; Development life cycle. System Conception: Devising a system concept; Elaborating a concept; Preparing a problem statement. Domain Analysis: Overview of analysis; Domain class model; Domain state model; Domain interaction model; Iterating the analysis.</p>			
Module-3: 10 Hrs			

Application Analysis: Application interaction model; Application class model; Application state model; Adding operations. Overview of system design; Estimating performance; Making a reuse plan; Breaking a system in to sub-systems; Identifying concurrency; Allocation of sub- systems; Management of data storage; Handling global resources; Choosing a software control strategy; Handling boundary conditions; Setting the trade-off priorities; Common architectural styles; Architecture of the ATM system as the example.

Module- 4 : 10 Hrs

Class Design: Overview of class design; Bridging the gap; Realizing use cases; Designing algorithms; Recursing downwards, Refactoring; Design optimization; Reification of behavior; Adjustment of inheritance; Organizing a class design; ATM example. Implementation Modeling: Overview of implementation; Fine-tuning classes; Fine-tuning generalizations; Realizing associations; Testing. Legacy Systems: Reverse engineering; Building the class models; Building the interaction model; Building the state model; Reverse engineering tips; Wrapping; Maintenance.

Module-5 : 12 Hrs

What is a pattern and what makes a pattern? Pattern categories; Relationships between patterns; Pattern description; Communication Patterns: Forwarder-Receiver; Client-Dispatcher-Server; Publisher-Subscriber; Management Patterns: Command processor; View Handler; Idioms: Introduction; What can idioms provide? Idioms and style; Where to find Idioms; Counted Pointer example.

Text Books:

1. Michael Blaha, James Rumbaugh: Object-Oriented Modeling and Design with UML, 2nd Edition, Pearson Education, 2005. (Chapters 1 to 17, 23)
2. 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.(Chapters 1, 3.5, 3.6, 4)

Subject Name: Data Mining

Subject Code	17MCA52	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	52	SEE Hours	03

CREDITS – 04

Module -1 :10 Hrs.

Data Warehouse

What is Data warehouse?, Differences between operational database systems, A multidimensional data model: data cubes, stars, snowflakes and fact constellations, Concept hierarchies, OLAP operations in multidimensional data model, Data warehouse architecture – steps for the design and construction of data warehouses, a three tier data warehouse architecture, types of OLAP servers, Data warehouse implementation – efficient computational of data cubes, Indexing OLAP data. What motivated data mining? What is data mining?

Module-2:12Hrs

Data Preprocessing

Why preprocess data, Descriptive data summarization – measuring the central tendency, dispersion of data, graphic displays of basic descriptive data summaries, Data cleaning - missing values, noisy data, data cleaning as process, Data integration and Transformation , data reduction data cube aggregation, attribute subset selection, dimensionality reduction, numerosity reduction Data discretization and concept hierarchy generations .

Module-3:12Hrs

Classification and Prediction

What is classification and prediction?, issues regarding classification and prediction, classification by decision tree induction – attribute selection method – information gain, gain ratio, Gini index, Tree pruning, Lazy learners- k-nearest-neighbor classifiers, case based reasoning, Prediction Accuracy and error measures- classifier accuracy measure, predictor error measures, Evaluating the accuracy of a classifier or predictor – holdout method and random sub sampling, cross validation, bootstrap.

Module-4: 09 Hrs

Market basket analysis, frequent itemsets, closed itemsets, and association rules , Association Rule mining: Association Rule Mining Basic concepts, frequent item sets, The Apriory Algorithm, Generating association rules from frequent item sets. Association Rule Mining without candidate generation -using FP-growth algorithm, Mining multilevel association rules, mining multidimensional association rules.

Module-5:09 Hrs.**Clustering**

Clustering: What is Cluster analysis? Types of data in cluster analysis- interval-scaled variables, binary variables, categorical, ordinal, ratio-scaled variables, variables of mixed types, vector objects, A categorization of major clustering methods, Partitioning Methods: The K-means method and k-medoids method Hierarchical methods: Agglomerative and Divisive hierarchical clustering, Dendrogram, Measures for distance between clusters: Minimum distance, maximum distance, average distance (Clusters Problems using these measures).

Text Books:

1. Data Mining Concepts and Techniques- Jiawei Han & Micheline Kamber Morgan Kaufmann Publishers – Second Edition) Unit I- Chapter 1 and 3 – (1.1, 1.2, 3.1.1, 3.1.2, 3.2.1, 3.2.2, 3.2.3, 3.2.5, 3.2.6, 3.3.1, 3.3.2, 3.3.5, 3.4.1, 3.4.2, 3.4.3) Unit II – Chapter 2: (2.1, 2.2, 2.3, 2.4, 2.5, 2.6) Unit III – Chapter 6: (6.1, 6.2, 6.3.1, 6.3.2, 6.3.3, 6.9.1, 6.9.2, 6.11.1-6.11.36.12, 6.13) Unit IV Chapter 5: (5.1.1, 5.1.2, 5.1.3, 5.2.1, 5.2.2, 5.2.4, 5.3.1, 5.3.2) Unit V - Chapter 7: (7.1, 7.2.1-7.2.5, 7.3, 7.4.1, 7.5.1).

References:

1. Introduction to Data mining – Pang-Ning Tan, Michael Steinbach, Vipin Kumar, Pearson Education.

Subject Name: Computational Intelligence

Subject Code	17MCA53	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	52	SEE Hours	03

CREDITS – 04

Module-1: 08Hrs

Introduction, Definitions, biological basis for neural nets, biological versus artificial neural nets, chromosomes, biological versus artificial chromosomes, behavioral motivations for fuzzy logic, Myths about computational intelligence, neural networks, evolutionary computation, fuzzy logic.

Module--2: 08 Hrs

Computational intelligence, adaptation, adaptation versus learning, three space adaptation, self-organization and evolution, evolution beyond Darwin, Historical views, Computational intelligence as adaptation and self-organization, ability to generalize. Soft computing versus hard computing

Module--3: 12Hrs

Evolutionary computation concepts and paradigms, History of evolutionary computation, genetic algorithms, evolutionary programming, genetic programming, particle swarm optimization, towards unification, evolutionary computation overview, overview of genetic algorithms, GA problem, GA operations, schemata and schema. Particle swarm optimization.

Module-- 4 : 12 Hrs

Neural network paradigms and concepts, Neural network history, components and terminology, neural network topologies, neural network adaptation, comparing neural networks and other information processing models, Kalman filters, correlation, radial basis functions. Preprocessing and post processing.

Module--5 : 12 Hrs

Neural network implementation, topology, back-propagation network initialization, learning vector quantization, BPN supervised adaptation, LVQ adaptation, issues in evolving neural networks, BPN implementation, Kohonen feature maps , implementation,

Text Book:

1. Eberhart, Computational Intelligence, Elsevier India publications.

Reference Book:

1. David poole, Allan Macworth, Randy Goebel, Computational Intelligence – A Logical Approach, Oxford University Press, New York.

Subject Name: Fuzzy Logic and Applications

Subject Code	17MCA54	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	52	SEE Hours	03

CREDITS – 04

Module 1:12 hours

INTRODUCTION: Historical perspective, utility of fuzzy systems, limitations of fuzzy systems, statistics and random processes, uncertainty in information, fuzzy sets and membership, chance versus fuzziness, sets as points in Hypercube.

CLASSICAL SETS AND FUZZY SETS: classical sets, operations on them, mapping of classical sets to functions, fuzzy sets, fuzzy set operations, properties of fuzzy sets, non-interactive fuzzy sets.

CLASSICAL RELATIONS AND FUZZY RELATIONS: Cartesian Product, Crisp Relations - Cardinality of Crisp Relations, Operations on Crisp Relations, and Properties of Crisp Relations, Composition. Fuzzy Relations - Cardinality of Fuzzy Relations, Operations on Fuzzy Relations, Properties of Fuzzy Relations, Fuzzy Cartesian Product and Composition, No interactive Fuzzy Sets. Tolerance and Equivalence Relations – Crisp Equivalence Relation, Crisp Tolerance Relation, Fuzzy Tolerance and Equivalence Relations. Value Assignments - Cosine Amplitude, Max-min Method, Other Similarity methods.

Module-2:12 Hours

MEMBERSHIP FUNCTIONS: Features of the Membership Function, Standard Forms and Boundaries, Fuzzification, defuzzification to crisp sets, Lambda-Cuts for Fuzzy Sets, Lambda-Cuts for Fuzzy Relations, Defuzzification Methods.

Development of membership Functions: Membership value assignments, intuition, inference, rank ordering, neural networks, genetic algorithms, inductive reasoning.

Module-3:12 Hours

FUZZY ARITHMETIC AND THE EXTENSION PRINCIPLE - Crisp Functions, Mapping and Relations, Functions of fuzzy Sets – Extension Principle, Fuzzy Transform (Mapping), Practical Considerations. Fuzzy Numbers Interval Analysis in Arithmetic, Approximate Methods of Extension - Vertex method, DSW Algorithm, Restricted DSW Algorithm, Comparisons. Fuzzy Vectors.

LOGIC AND FUZZY LOGIC: Classical Predicate Logic –Tautologies, Contradictions, Equivalence, Exclusive Or Exclusive Nor, Logical Proofs, Deductive Inferences. Fuzzy Logic, Approximate Reasoning, Fuzzy Tautologies, Contradictions, Equivalence and Logical Proofs, Other forms of the Implication Operation, Other forms of the Composition Operation.

Module- 4 :08 hours

FUZZY RULE BASED SYSTEMS: Natural Language, Linguistic Hedges, Rule- Based Systems - Canonical Rule Forms, Decomposition of Compound Rules, Likelihood and Truth Qualification, Aggregation of Fuzzy Rules. Graphical Techniques of Inference.

Module -5 :8 Hours

FUZZY CLASSIFICATION: Classification by equivalence relations, fuzzy relations, cluster analysis, cluster validity, C-means clustering, hard-c means(HCM), fuzzy c-means(FCM), fuzzy c-means algorithm, classification metric, hardening the fuzzy partition and similarity relations from clustering.

Text Book:

Timothy J. Ross, Fuzzy Logic with Engineering Applications, Wiley India, II Edition, 2010 reprint. Chapters 1,2,3,4,5,6,Chapter 11 (Part I only), 12.

Reference Books:

1. John Yen, Reza Langari, Fuzzy Logic- Intelligence, Control, and information, Pearson Education, 2004.
2. George J.Klir , Bo Yuan, Fuzzy Sets and Fuzzy Logic-Theory and Applications, Prentice Hall of India, 2000.

Subject Name: Digital Image Processing

Subject Code	17MCA55	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	52	SEE Hours	03
CREDITS – 04			

Module-1: 12 Hours

DIGITAL IMAGE FUNDAMENTALS:

Origin of Digital Image processing – fundamental steps –Components of Image Processing system –Visual perception –Light and EM spectrum –Image sensing and acquisition –Image sampling and Quantization – relationship between pixels, Two-Dimensional Mathematical Preliminaries.

Module -2:10 Hours

IMAGE ENHANCEMENT:

Spatial Domain: Gray level transformation –Histogram processing –Arithmetic / Logic operations- Spatial filtering –smoothing filters – sharpening filters Frequency Domain: Fourier transform –smoothing frequency domain filters –sharpening filters –Homographic filtering.

Module – 3 :10 Hours

IMAGE RESTORATION: Image Restoration - Degradation Model, Unconstrained Restoration - Lagrange multiplier and constrained restoration, Inverse filtering removal of blur caused by uniform linear motion, Wiener filtering, Geometric transformations-spatial transformations.

Module – 4:10 Hours

IMAGE SEGMENTATION:

Edge detection, Edge linking via Hough transform, Thresholding, Region based segmentation, Region growing, Region splitting and Merging, Segmentation by morphological watersheds, Basic Concepts, Dam Construction, Watershed segmentation algorithm.

Module 5: 10 Hours

IMAGE COMPRESSION:

Need for data compression, Fundamentals –Image compression models Huffman, Run Length Encoding, Shift codes, Arithmetic coding, Vector Quantization, Transform coding, JPEG standard, MPEG.

Reference

1. Rafael C. Gonzalez, Richard E. Woods, "Digital Image Processing", Pearson, Second Edition, 2004 Anil K. Jain," Fundamentals of Digital Image Processing", Pearson 2002.Kenneth R. Castleman," Digital Image Processing", Pearson, 2006.
2. Rafael C. Gonzalez, Richard E. Woods, Steven Eddins,"Digital Image Processing using MATLAB", Pearson Education, Inc., 2004.
3. D.E. Dudgeon and RM. Mersereau, "Multidimensional Digital Signal Processing", Prentice Hall Professional Technical Reference, 1990.
4. William K. Pratt, "Digital Image Processing" , John Wiley, New York, 2002 Milan Sonka et al, "IMAGE PROCESSING, ANALYSIS AND MACHINE VISIO", Brookes/Cole, Vikas Publishing House, 2nd edition, 1999.

Subject Name: Data Mining Lab			
Subject Code	17MCA56	CIE Marks	10
Number of Practical Hours/Week	04	SEE Marks	40
Total Number of Lecture Hours	52	SEE Hours	03
CREDITS – 02			

List of Programs using R

1. Data Exploration and visualization with R
2. Regression with R
3. Classification with R
4. Data Clustering with R
5. Association Rule Mining with R

Subject Name: Software Design Lab

Subject Code	17MCA57	CIE Marks	10
Number of Practical Hours/Week	04	SEE Marks	40
Total Number of Lecture Hours	52	SEE Hours	03
CREDITS – 02			

Lab Programs

The student has to draw the necessary UML diagrams using any suitable UML Drawing Tool and implement in Java OR C++ OR C# a program to demonstrate the Design Pattern specified by the Examiner.

The Design Pattern is allotted based on lots from the following list:

- 1) Expert
- 2) Controller
- 3) Publisher-Subscriber
- 4) Command
- 5) Forward-Receive
- 6) Client-Dispatcher
- 7) Proxy
- 8) Façade
- 9) Polymorphism

Subject Name: Mini Project -II

Subject Code	17MCA58	CIE Marks	10
Number of Practical Hours/Week	04	SEE Marks	40
Total Number of Lecture Hours	52	SEE Hours	03
CREDITS – 04			

Mini project which are oriented to topics not restricted to DBMS, Image processing, Wireless sensor network, data mining, data analytics is to be developed by each batch consisting of a maximum of two students.

SEMESTER-VI

Subject Name: E-GOVERNANCE			
Subject Code	17MCA61	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	52	SEE Hours	03
CREDITS – 04			
<p>Module-1:12 Hours Introduction to e- Governance, Different Stages of e-Governance, Advantages, Problems and Challenges of e-Governance, National Statues, International Status, Securities in e-Governance.</p> <p>Module– 2:10 Hours National e-Governance Plan, Government of India guidelines for websites, W3C guidelines, web 2.0, web 3.0</p> <p>Module– 3:10 Hours Different UN Survey on e-Governance, UN Survey on e-Governance – 2014, e-Government Act, 2002, Adhaar Bill, 2016, II Administrative Reforms Committee Report 11, Digital India Programme, IT Act, 2008 Section 1 to 11A, Section 43</p> <p>Module– 4:10 Hours Workflow Management in e-Governance, Digital Divide, Mechanism to handle Digital Divide, Bridge the digital divide, M-Governance, e-Learning, Role of Social Media in e-Governance, Big data Analytics in e-Governance, Semantic web Analytics</p> <p>Module-5:10 Hours Case Study: Election Commission, Indian Railway Reservation, Aadhar – UID, Income Tax, SAKALA, Bhoomi, e-Commission, CET admission, Centralized Admission, Student Scholarship Management.</p>			
<p>Reference</p> <ol style="list-style-type: none"> 1. Mishra D.S (2007). E-Governance as reform strategy for combating corruption in delivery of public services. Indian Journal of Public Administration. LIII (3). 2. Bhogle Srinivas (2009). E-Governance. Selected Readings on Information Technology Management: Contemporary Issues ed. George Kelley. Information Science Reference, New York. 			

3. Bhuiyan H Shahjahan (2011). Modernizing Bangladesh public administration through e-governance: Benefits and challenges. 28, 54-65.
4. The World Wide Web Consortium (2008). Web Content Accessibility Guidelines (WCAG)
5. 2.0. Downloaded on 10th January, 2012 from <http://www.w3.org/>
6. Government of India (2009). Guidelines for Indian Government websites. Downloaded on 15th January, 2012 from <http://darpg.nic.in/>
7. e-Government Act (2002). <https://www.gpo.gov/fdsys/pkg/PLAW-107publ347/pdf/PLAW-107publ347.pdf>
8. Digital India Programme. <http://www.digitalindia.gov.in/>

Subject name: Parallel Algorithms

Subject Code	17MCA62	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	52	SEE Hours	03

CREDITS – 04

Module 1:12 Hours

INTRODUCTION:

Introduction to Parallel Algorithms – Models of Parallel Computation – Sorting on an EREW-SIMDPRAM Computer – Relation between PRAM Models – SIMD Algorithms – MIMD Algorithms – Selection – Desirable Properties for Parallel Algorithms - Parallel Algorithm for Selection – Analysis of Parallel Algorithms.

Module -2 :10 Hours

SORTING AND SEARCHING:

Merging on the EREW and CREW Models - Fast Merging on EREW - Sorting Networks – Sorting on a Linear Array – Sorting on CRCW, CREW, EREW Models – Searching a Sorted Sequence – Searching a Random Sequence.

Module- 3 :10 Hours

ALGEBRAIC PROBLEMS:

Generating Permutations and Combinations in Parallel – Matrix Transpositions – Matrix by Matrix Multiplications – Matrix by Vector multiplication.

Module- 4:10 Hours

GRAPH THEORY AND COMPUTATIONAL GEOMETRY PROBLEMS:

Connectivity Matrix – Connected Components – All Pairs Shortest Paths – Minimum Spanning Trees – Point Inclusion – Intersection, Proximity and Construction Problems - Sequential Tree Traversal - Basic Design Principles – Algorithm – Analysis

Module - 5 10 Hours

DECISION AND OPTIMIZATION PROBLEMS:

Computing Prefix Sums – Applications - Job Sequencing with Deadlines – Knapsack Problem-
The Bit Complexity of Parallel Computations

Reference

1. Selim G. Akl, “The Design and Analysis of Parallel Algorithms”, Prentice Hall, New Jersey, 2009.
2. Michael J. Quinn, “Parallel Computing: Theory & Practice”, Tata McGraw Hill Edition, 2013.
3. Justin R. Smith, “The Design and Analysis of Parallel Algorithms”, Oxford University Press, USA , 2003.

SUBJECT NAME:ADVANCED DATABASE MANAGEMENT SYSTEMS

Subject Code	17MCA63	CIE Marks	20
Number of Lecture Hours/Week	04	SEE Marks	80
Total Number of Lecture Hours	52	SEE Hours	03

CREDITS – 04

Module-1:12 Hours

NOSQL and Query Optimization: Definition of NOSQL, History of NOSQL and Different NOSQL products, Exploring Mongo DB Basics: NOSQL Storage architecture, CRUD operations with Mongo DB, Querying, Modifying and Managing NOSQL Data stores, Indexing and ordering data sets (Mongo DB/Couch DB/Cassandra). Advanced NOSQL, NOSQL in CLOUD, Parallel Processing with Map Reduce, Big Data with Hive. Working with NOSQL:, Query Optimization: Overview, Transformation of Relational Expressions, Estimating Statistics of Expression Choice of Evaluation Plans, Materialized views Advanced Query Optimization: Motivation, Query Processing Phases, Logical Query Optimization.

Module-2:10 Hours

SAN: Introduction to Information Storage and Management, Data Center Infrastructure, Information Lifecycle Components of Storage System Environment, Disk Drive Components, Disk Drive Performance, Fundamental Laws Governing Disk Performance, Logical Components of the Host, Application Requirements and Disk Performance. Data Protection, Intelligent Storage system: Implementation of RAID, RAID Array Components, RAID Levels, RAID Impact on Disk Performance

Module-3:10 Hours

Data Warehousing and Data Mining: Data Warehouse Architecture, Data Warehouse Implementation, Mining Methods, Mining Various Kinds of Association Rules. Data Mining: Data Mining Applications, Social Network Analysis

Module-4:10 Hours

Big Data: Introduction to principles and practice of systems that improve performance through experience. Topics include statistical learning framework, supervised and unsupervised learning, performance evaluation and empirical methodology; design tradeoffs. Introduction to the Big Data problem. Current challenges, trends, and applications Algorithms for Big Data analysis. Mining and learning algorithms that have been developed specifically to deal with large datasets Technologies for Big Data management. Big Data technology and tools, special consideration made to the Map Reduce paradigm and the Hadoop ecosystem

Module-5:10 Hours

Information Retrieval and Search Engines: Architecture of search engine, Ranking and Evaluation; CRAWLS AND FEEDS: Crawling the Web, Directory Crawling, Conversion Problem, Storing the Documents, Detecting Duplicates. Processing text: Text Statistics, Document Parsing, Document Structure and Markup, Link analysis, Information Extraction, Internationalization; RANKING WITH INDEXES: Inverted indexes, Compression, Entropy and Ambiguity, Delta Encoding, Bit-aligned codes, Auxiliary Structures, Index Construction, Query Processing

Reference

1. “Professional NOSQL” by Shashank Tiwari, 2011, WROX Press The Definitive guide to MongoDB, The NoSQL Database for Cloud and Desktop Computing, by Eelco Plugge, Tim Hawkins, Peter Membrey Apress 2010 "NoSQL Handbook” by Mathias Meyer, 2011 Paperplanes.
2. MongoDB: The Definitive Guide, 2nd Edition, by Kristina Chodorow 2013 Silberschatz, Korth and Sudharshan Andreas Meister Otto-von-Guericke University Magdeburg
3. G. Somasundaram, Alok Shrivastava (Editors): Information Storage and Management: Storing, Managing & Protecting Digital Information in Classic, Visualized and Cloud Environments, 2 nd edition, EMC Education Services, Wiley- India, 2009. ISBN 978-1-1180-9483-9
4. Jiawei Han and Micheline Kamber, Data Mining, Concepts and Techniques, Morgan Kaufmann Publisher, II Edition, 2006.
7. Machine Learning, Tom Mitchell. ISBN-10: 0070428077 | ISBN-13: 978- 0070428072 | Editi993on: 1 (optional)

Subject Name: Dissertation/ Project			
Subject Code	17MCA64	CIE Marks	100
Number of Practical Hours/Week	28	SEE Marks	250
Total Number of Lecture Hours	-	SEE Hours	03
CREDITS – 14			

- a) Is spread over VI semester and evaluated at the end of VI semester for the assigned credits.
- b) The project may be based on;
 - Design of a system.
 - Theoretical/Analytical modeling.
 - Computer Simulation.
 - Developing working software
 - Interdisciplinary computer application/modeling.
- c) The project could be part of the research activity carried out in the department.
- d) The literature survey should be one of the components of the project.
- e) The project can be carried outside the institute in a recognized industry/research lab.
- f) Director and DAAC assign guides for the major project.
- g) The project is to be carried out by individual student.

CIE for the major project

Comprises of three seminars. Departmental Project Evaluation Committee (DPEC) shall evaluate seminars along with the respective guides.

- In the Project Preview-1, students have to present about the area of the project, literature survey and preliminary requirements of the project and what they propose to do.
- In the Project Preview -2, the students have to present about the project development issues like, specifications, flow chart, design steps, data flow diagrams, data structures, entity relationship diagrams pertaining to the chosen project.

- In the Project Preview -3, the students have to present a demo of the project.
- CIE is done for a total of 100 marks,

The breakup of marks for CIE is given in Table 1.

Table 1. Break up of CIE marks for major project

Project Preview-1	20 marks
Project Preview -2	40 marks
Project Preview -3	40 marks
Total	100 marks

Table 2. Break up of SEE marks for major project

Evaluation of the project report by external examiner and the guide (average of independent evaluations)	100
Presentation, Demonstration and Quality of work	100
viva-voce	50
Total	250

The End