JSS Mahavidyapeetha

JSS Science and Technology University

Sri Jayachamarajendra College of Engineering

Mysuru - 570 006.

Department of Master of Computer Applications

OVERALL COURSE GRID OF

I to VI Semesters

2016-2019

Scheme of Teaching and Examination

First Semester MCA

SL. No.	Subject Code	Course Title	Teaching Department	Credits				Contact Hours	Marks			Exam Duration (Hrs)
				L	T	P	Total		CIE	SEE	Total	
1.	MCA110	Management Information Systems	MCA	4	0	0	4.0	4	50	50	100	3
2.	MCA120	Discrete Mathematical structures	MCA	4	0	0	4.0	4	50	50	100	3
3.	MCA130	Computer Organization & Architecture	MCA	4	0	0	4.0	4	50	50	100	3
4.	MCA140	Problem Solving using C	MCA	4	0	0	4.0	4	50	50	100	3
5.	MCA150	Introduction to Linux	MCA	4	0	0	4.0	4	50	50	100	3
6.	MCA160	Soft skills-I (Professional Communications)	MCA	2	0	0	2.0	2	50	50	50	11/2½
6.	MCA13L	Digital Electronics Lab	MCA	0	0	1	1	3	50	00	50	0
7.	MCA14L	C programming Lab	MCA	0	0	1	1	3	50	00	50	0
8.	MCA15L	Linux Lab	MCA	0	0	1	1	3	50	00	50	0
Total				22	0	3	25	31	450	300	700	-

Scheme of Teaching and Examination

Second Semester MCA

SL. No.	Subject Code	Course Title	Teaching Department	Credits				Contact Hours	Marks			Exam Duration (Hrs)
				L	T	P	Total		CIE	SEE	Total	
1.	MCA210	System Software	MCA	4	0	0	4.0	4	50	50	100	3
2.	MCA220	Operating System	MCA	4	0	0	4.0	4	50	50	100	3
3.	MCA230	Data Structures with C++	MCA	4	0	0	4.0	4	50	50	100	3
4.	MCA240	Software Engineering-I	MCA	4	0	0	4.0	4	50	50	100	3
5.	MCA250	Computer Networks-I	MCA	4	0	0	4.0	4	50	50	100	3
6.	MCA260	Soft skills –II (Ethics in IT)	MCA	2	0	0	2.0	2	50	50	50	11/2½
7.	MCA21L	System Software Lab	MCA	0	0	1	1	3	50	00	50	-
8.	MCA22L	Data structures with C++ Lab	MCA	0	0	1	1	3	50	00	50	-
9.	MCA25L	Computer Networks Lab	MCA	0	0	1	1	3	50	00	50	
Total				22	0	3	25	31	450	300	700	-

Scheme of Teaching and Examination

Third Semester MCA

SL. No.	Subject Code	Course Title	Teaching Department	Credits				Contact Hours	Marks			Exam Duration (Hrs)
				L	T	P	Total		CIE	SEE	Total	
1.	MCA310	Relational Database Management Systems	MCA	4	0	0	4.0	4	50	50	100	3
2.	MCA320	Java Programming	MCA	4	0	0	4.0	4	50	50	100	3
3.	MCA330	Software Engineering - II	MCA	4	0	0	4.0	4	50	50	100	3
4.	MCA340	Analysis & Design of Algorithms	MCA	4	1	0	5.0	6	50	50	100	3
5.	MCA350	Computer Networks -II	MCA	4	1	0	5.0	6	50	50	100	3
6.	MCA360	Soft Skills – III (Personal effectiveness and leadership)	MCA	2	0	0	2.0	2	50	50	50	11/2
7.	MCA31L	Relational Database Management Systems Laboratory	MCA	0	0	1	1	3	50	-	50	-
8.	MCA32L	Java Programming Laboratory	MCA	0	0	1	1	3	50	-	50	-
9.	MCA33L	Software Engineering Laboratory	MCA	0	0	1	1	3	50	-	50	-
Total				22	2	3	27	34	450	300	700	-

Scheme of Teaching and Examination

Fourth Semester MCA

SL. No.	Subject Code	Course Title	Teaching Department	Credits				Contact Hours	Marks			Exam Duration (Hrs)
				L	T	P	Total		CIE	SEE	Total	
1.	MCA410	Web Programming	MCA	4	0	0	4.0	4	50	50	100	3
2.	MCA420	Advanced Java Programming	MCA	4	0	0	4.0	4	50	50	100	3
3.	MCA430	Cloud Computing	MCA	4	1	0	5.0	6	50	50	100	3
4.	MCA44X	Elective-I	MCA	4	0	0	4.0	4	50	50	100	3
5.	MCA45X	Elective – II	MCA	4	0	0	4.0	4	50	50	100	3
6.	MCA460	Soft Skills – IV (Research Methodology & Seminar)	MCA	1	1	0	2.0	2	50	50	50	11/2½
7.	MCA41L	Web Programming Laboratory	MCA	0	0	1	1.0	3	50	-	50	-
8.	MCA42L	Advanced Java Programming Laboratory	MCA	0	0	1	1.0	3	50	-	50	-
Total				21	2	2	25	29	400	300	650	-

Elective –I

SL. No.	Subject Code	Course Title
1.	MCA441	Digital Image Processing
2.	MCA442	Advanced Algorithms
3.	MCA443	Distributed Operating Systems
4.	MCA444	Software Project Management

Elective – II

SL. No.	Subject Code	Course Title
1.	MCA451	Cryptography and Network Security
2.	MCA452	Principles of User Interface Design
3.	MCA453	Enterprise Resource Planning
4.	MCA454	Computer Forensics

Scheme of Teaching and Examination

Fifth Semester MCA

SL. No.	Subject Code	Course Title	Teaching Department	Credits				Contact Hours	Marks			Exam Duration (Hrs)
				L	T	P	Total		CIE	SEE	Total	
1.	MCA510	Programming with Python	MCA	4	0	0	4.0	4	50	50	100	3
2.	MCA520	Data Mining & Analytics	MCA	4	0	0	4.0	4	50	50	100	3
3.	MCA530	Software Architectures	MCA	4	0	0	4.0	4	50	50	100	3
4.	MCA54X	Elective – III	MCA	4	0	0	4.0	4	50	50	100	3
5.	MCA55X	Elective – IV	MCA	4	0	0	4.0	4	50	50	100	3
6.	MCA560	Soft Skills – V (Entrepreneurship)	MCA	2	0	0	2.0	2	50	50	50	11/2½
6.	MCA51L	Python Lab	MCA	0	0	3	1.0	3	50	-	50	-
7.	MCA52L	Data Mining & Analytics Lab	MCA	0	0	3	1.0	3	50	-	50	-
6.	MCA56P	Mini Project	MCA	0	0	3	1.0	3	50	-	50	-
Total				22	0	9	25	29	450	300	700	-

Elective – III

Sl. No	Subject Code	Course Title
1.	MCA541	Computer Vision & Pattern Recognition
2.	MCA542	Information Retrieval and Web Mining
3.	MCA543	Distributed Operating System
4.	MCA544	Operation Research
5.	MCA545	Big Data

Elective – IV

Sl. No.	Subject code	Course Title
1.	MCA551	Business Intelligence
2.	MCA552	Software Testing
3	MCA553	Network Management
4.	MCA554	. Net Programming
5.	MCA555	Supply Chain Management

Scheme of Teaching and Examination

Sixth Semester MCA

SL. No.	Subject Code	Course Title	Teaching Department	Credits				Weekly Contact Hours	Marks			Exam Duration (Hrs)
				L	T	P	Total		CIE	SEE	Total	
1.	MCA61P	Project Work	MCA	0	0	23	23.0	2	100	150	250	1
Total				0	0	23	23.0	2	100	150	250	-

Credit Pattern for MCA Course

Semester	Credits
1	25
2	25
3	27
4	25
5	25
6	23
Total	150

Course Year	Course Semester	Course Type	Credits				Total Hours/ Semester
				Theory	Laboratory	Tutorials	
I	I	Theory	04	04	0	0	52

Course No	Course Title	Pre Requisites
MCA110	Management Information Systems	NIL

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2 Events- Marks: 50].
2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, the student will be :

1. Able to understand the concepts of system, system components and development techniques with case studies.
2. Understanding the Design and develop Information system for business environment by understanding the need and criticality of type of information.
3. Able to apply the different strategies for the management of business to formulate business process.
4. Able to analyze the need for business process re-engineering, and the process of making.
5. Understanding the concepts and functionalities of enterprise systems like ERP, SCM, DSS.

TOPICS COVERED:

UNIT:1 - Systems Engineering & Information and Knowledge 10 Hours

System concepts, system control, types of systems, handling system complexity, Classes of systems, General model of MIS, Need for system analysis, System analysis for existing system & new requirement, system development model, MIS & system analysis. 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, Knowledge.

UNIT:2 - Introduction of MIS and Development of MIS 10 Hours

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. Development of long range plans of the MIS, Ascertaining the class of information, Determining the information requirement, Development and implementation of the MIS, Management of information quality in the MIS, Organization for development of MIS, MIS development process model.

UNIT: 3 - Strategic Management of Business & Developing Business/IT Strategies/ITSolutions 10 Hours

Concept of corporate planning, Essentiality of strategic planning, Development of the business strategies, Type of strategies, short-range planning, tools of planning, MIS: strategic business planning. Planning fundamentals (real world cases), Organizational planning, planning for competitive advantage, (SWOT Analysis), Business models and planning. Business/IT planning, identifying business/IT strategies, Implementation Challenges, Change management., Developing business systems, (real world case), SDLC, prototyping, System development process, implementing business system.

UNIT: 4 - Business Process Re-Engineering & Technology of Information System 10 Hours

Introduction, Business process, process model of the organization, value stream model of the organization, what delay the business process, relevance of information technology, MIS and BPR. Introduction, Data processing, Transaction processing, Application processing, information system processing, TQM of information systems, Human factors & user interface, Strategic nature of IT decision, MIS choice of information technology.

UNIT:5 - Decision Making and DSS, Electronic Business Systems & Client Server Architecture and E-business Technology: 12 Hours

Decision making concepts; decision making process, decision-making by analytical modeling, Behavioral concepts in decision making, organizational decision-making, Decision structure, DSS components, and Management reporting alternatives. Enterprise business system – Introduction, cross-functional enterprise applications, real world case, Functional business system, - Introduction, marketing systems, sales force automation, CIM, HRM, online accounting system, Customer relationship management, ERP, Supply chain management. Client server architecture, implementation strategies, Introduction to E-business, model of E business, internet and World Wide Web, Intranet/Extranet, Electronic, Impact of Web on Strategic management, Web enabled business management, MIS in Web environment.

TEXT BOOKS / REFERENCES:

1. Waman S Jhawadekar: Management Information System, 3rd Edition, Tata McGraw Hill. 4th Reprint 2012

2. James A O'Brien and George M Marakas: Management Information System, 7th Edition, Tata McGraw Hill, 12th Reprint 2011.

ADDITIONAL LEARNING SOURCES:

1. Ralph M Stair and George W Reynolds: Principles of Information Systems, 7th Edition, Thomson, 2010.
2. Steven Alter: Information Systems - The Foundation of E-Business, 4th Edition, Pearson Education, 4th Reprint 2011.
3. Mahadeo Jaiswal and Monika Mital: Management Information System, , Oxford University Press.
4. Effy Oz: Management Information Systems, 5th Edition, Thomson Course Technology,2010.
5. www.mu.ac.in/mis
6. http://www.tutorialspoint.com/management_information_system/mis_tutorial.

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week	Total Hours/ Semester

				Theory	Laboratory	Tutorials	
I	I	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA120	Discrete Mathematical Structures	NIL

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, Any 2 Events from the list [Test, Quiz, Assignments etc.] Marks: 50 [10* 5 Events].

2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, the student will be able to:

1. Define the syntax and semantics of propositional and predicate logic; translate statements from a natural language into its symbolic structures in logic.
2. Understand the concepts like Logic, Quantifiers, set theory, elementary and advanced counting techniques and permutations & combinations
3. Use pigeonhole principle, induction principle in proving statements about certain sets and Greatest Common Divisor.
4. Apply the notion of relations on some finite structures, like strings and databases; analyze algorithms using the concept of functions and function complexity.
5. Explore the techniques of Graph, Coding theory, Hamming Metric, the Parity-check and generator matrix.

TOPICS COVERED:

UNIT:1 - Fundamentals of Logic: 12 Hours

Basic Connectives and Truth Tables, Logic Equivalence :The laws of Logic, Logical Implications: Rules of Inference, The use of Quantifiers, Quantifier Definitions, Proofs of Theorems.

UNIT:2 - Set Theory 9 Hours

Sets and Subsets, Set Operations and the Laws of Set Theory, Counting and Venn Diagrams, Principles of Inclusion and Exclusion, The rules of sum and product, Permutations and Combinations with repetition

UNIT:3 - Properties of Integers and Recurrence: 9 Hours

Mathematical Induction, Recursive definitions, the first order Linear recurrence relation. The Greatest Common

Divisor Euclidian Algorithms

UNIT:4 - Relations and Functionsn **11 Hours**

Cartesian products and Relations, Functions-Plain and One-to-One, Onto Functions, Stirling Numbers and the Second Kind, Special functions, The Pigeon-hole principle, Function composition and inverse functions. Properties of Relations, Computer recognition-Zero One Matrices and Directed graphs, Posets and Hasse Diagrams, Equivalence relation and Partitions, lattices.

UNIT:5 - Graph Theory and Coding Theory **11 Hours**

Terminology, Definitions, Properties and Examples, Connectivity and Adjacency, Euler and Hamilton, Representation and Isomorphism, Planarity and Chromatic Number, Elements of Coding Theory, The Hamming Metric, The Parity Check Matrix and Generator Matrix.

TEXT BOOKS / REFERENCES:

1. Ralph P Grimaldi, B.V.Ramana, "Discrete & Combinatorial Mathematics, An Applied Introduction" 5th Edition, Pearson Education, 2004.
2. Eric Gosset "Discrete Mathematics with Proof" Wiley India, 2nd Edition, 2009.
3. Kenneth H Rosen, "Discrete Mathematics & its Applications" 7th edition, McGraw-Hill, 2010.
4. Y N Singh "Discrete Mathematical Structures" Wiley India, 1st edition, 2010.
5. Jayant Ganguly: A Treatise on Discrete Mathematical Structures" Pearson, 2010.

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	

I	I	Theory	04	04	00	00	52

Course No	**Course Title**	**Pre Requisites**
MCA130	Computer Organization and Architecture	NIL

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, Any 2 Events from the list [Test, Presentation, Quiz. Projects, Group discussion, assignments, Viva-Voce etc.] Marks: 50 [10* 5 Events].
2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, the student will be able to:

1. Design networks of logic gates which have a specified relationship between signals at the input and output terminals
2. Able to use the logical properties of flip-flops, which serve as memory devices. Understanding of interconnections in a logic network
3. Knowing the fundamental principles for computer architectures and organizations and machine instructions and able write small programs using mnemonics.
4. Knowledge on input output interface circuits , Use of interrupts and various bus mechanisms.
5. Knowledge on Various Memory management concepts like Cache Memories and Virtual Memories. Designing fast adders. Floating point representation on 32 bit & 64 bit architectures.

TOPICS COVERED:

UNIT-1 **16 Hours**

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 Gates, The NAND Gate as an universal Gate, Gates with More than two inputs, Using Inverters to convert gates.

Combining Logic Gates and Arithmetic Circuits: Constructing Circuits from Boolean Expression, Drawing a circuit from a Maxterm Boolean Expression, Truth Tables and Boolean Expressions, Sample Problem Simplifying Boolean Expression, Karnaugh Maps, Karnaugh Maps with three variables, Karnaugh Maps with four variables, more Karnaugh Maps, using Demorgan's Theorem, Binary Addition, Half Adders, Full Adders, Three Bit Adders, Binary Subtraction, Parallel Subtractors, Binary Multiplication, Binary Multipliers, 2s Complement Notation, Addition and Subtraction, 2s Complement adders/subtractor.

UNIT:2 **10 Hours**

Basic Structure of Computer: Computer Types, Functional Units, Basic Operational Concepts, Bus

structures, Performance, Memory Location and Addresses. **Machine Instruction and Programs:** Memory Operations, Instructions & Instruction Sequencing, Addressing Modes, Assemble Language, Basic Input/Output Operations

UNIT:3 **10 Hours**

Input/Output Organization: Accessing I/O Devices, Interrupts, Direct Memory Accesses, Buses, Interface Circuits.

UNIT:4 **10 Hours**

Memory Systems: Some Basics concepts, Semiconductors RAM Memories, Read-Only Memories, Cache Memories, Virtual Memories. **Arithmetic Unit:** Addition & subtraction of Signed Numbers, Design of Fast adders, Multiplication of Positive Numbers, Floating-Point Numbers & Operations

UNIT:5 **6 Hours**

Machine Architecture: Introduction, System Software and Machine Architecture, Simplified Instructional Computer (SIC) - SIC Machine Architecture, SIC/XE Machine Architecture, SIC Programming Examples.

TEXT BOOKS / REFERENCES:

a. Tokheim : Digital Electronics principles and applications, 6th edition, <cgraw hill , 2012.
b. Carl Hamacher, Z. varnesic and S.Zaky : Computer Organization, 5th edition, Mcgraw Hill, 2010
c. Leland L beck, D Manjula: System Software, 3rd edition, pearson education, 2010.

ADDITIONAL LEARNING SOURCES:

1. www.tutorialspoint.com/computer_logical_organization/.
2. www.freetechbooks.com/computer-organization-and-architecture-f56.html

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
I	I	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA140	Problem Solving Using C	NIL

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, Any 2 Events from the list [Test, Presentation, Quiz] Marks: 50 [10* 5 Events].
2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, the student will be able to:

1. Able to have fundamental knowledge on basic programming aspects and storage classes.

2. Able to understand the basic terminologies used in computer programming, write, compile and debug programs in C language.
3. Able to use different data types in a computer program.
4. Able to design programs involving decision structures, loops and functions.
5. Able to understand the dynamics of memory by the use of pointers.

TOPICS COVERED:

UNIT 1: **9 Hours**

Algorithms, Flow Charts, C structure, Variables, Data types, Constants, Declarations, Operators, Precedence, Associativity, Order of evaluation, Type conversion, Storage classes, Programming Examples

UNIT 2: **10 Hours**

Input and output statements – scanf, getchar, gets, printf, putchar, puts; Control Statements – if, else-if, switch, Control Structures – while, for, do-while, break and continue, Examples

UNIT 3: **9 Hours**

Arrays – Single dimension, Two dimensional, Multi dimensional Arrays, Strings, Programming. Functions, Categories of functions. Examples

UNIT 4: **12 Hours**

Pointers, Pointer arithmetic, Call by value, Pointer Expression, Pointer as function arguments, , recursion, Passing arrays to functions, passing strings to functions, Call by reference, Functions returning pointers, Pointers to functions, Programming Examples
Structures and Unions – defining, declaring, initialization, accessing, comparing, operations on individual members; array of structures, structures within structures, structures and functions, pointers and structures, bit fields, Programming Examples

UNIT 5: **12 Hours**

Files – defining, opening, closing, input and output operations, error handling, random access; Command line arguments; Dynamic Memory Allocation –definition, malloc, calloc, realloc, free, dynamic arrays; Preprocessor – definition, macro substitution, file inclusion, compiler control directives, Programming Examples.

TEXT BOOKS / REFERENCES:

1. Programming in ANSI C, Third Edition, E.Balaguruswamy. 6th Edition (2013)
2. The C Programming Language, Brian W Kernighan, Dennis M Rtchie, PHI, 2nd Edition.
3. Let us C, 4th Edition, Yashwanthkanetkar.
4. Programming with C, Byron Gottfried, Tata McGraw-Hill edition.

ADDITIONAL LEARNING SOURCES:

1. http://www.worldbestlearningcenter.com/index_files/c_tutorial_lesson.htm
2. http://www.cluster2.hostgator.co.in/files/writeable/uploads/hostgator99706/file/letusc-yashwantkanetkar.pdf

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
I	I	Theory	05	04	01	00	52

Course No	Course Title	Pre Requisites
MCA150	**Introduction to LINUX**	Computer Concepts

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, Any 2 Events from the list [Test, Presentation, Quiz. Projects, Group discussion, assignments, Viva-Voce etc.] Marks: 50 [10* 5 Events].

2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, the student will be able to:

1. Get the working Knowledge of an operating System
2. Comfortably work in UNIX operating system Environment.
3. Get the knowledge of SHELL and write simple SHELL programs.
4. Get the Knowledge of UNIX file system and Utilities available for file handing.
5. Able to use simple and advanced filters.

TOPICS COVERED:

Unit 1: 10 Hours

The Operating System, The UNIX Operating System, Knowing Your Machine, A Brief Session, How It All Clicked, Linux and GNU, The UNIX Architecture, Features of UNIX, POSIX and the Single UNIX Specification, Locating Commands, Internal and External Commands, Command Structure , Flexibility of Command Usage, Man Browsing the Manual Pages On-line, Understanding the man Documentation , Further Help with **man -k**, apropos and **whatis**, When Things Go Wrong, **cal**: The Calendar, **date**: Displaying the System Date, **echo**: Displaying a Message, **printf**: An Alternative to echo, **bc**: The Calculator, **script**: Recording Your Session, Email Basics, **mailx**: The Universal Mailer, **passwd**: Changing Your Password , **who**: Who Are the Users?, **uname**: Knowing Your Machine's Characteristics, **tty**: Knowing Your Terminal, **stty:** Displaying and Setting Terminal Characteristics.

Unit 2: 10 Hours

The File, What's in a (File) name? , The Parent-Child Relationship, The HOME Variable: The Home Directory, **pwd:** Checking Your Current Directory, **cd:** Changing the Current Directory, **mkdir:** Making Directories, **rmdir:** Removing Directories, Absolute Pathnames, Relative Pathname, **ls:** Listing Directory Contents, The UNIX File System, **cat:** Displaying and Creating Files, **cp:** Copying a File, **rm:** Deleting Files, **mv:** Renaming Files, **more:** Paging Output, The lp Subsystem: Printing a File, **file:** Knowing the File Types, **wc:** Counting Lines, Words and Characters, **od:** Displaying Data in Octal, **cmp:** Comparing Two Files, **comm:** What is Common?, **diff:** Converting One File to Other, **dos2unix** and **unix2dos**: Converting Between DOS and UNIX, Compressing and Archiving Files, **gzip** and **gunzip**: Compressing and Decompressing Files, **tar:** The Archival Program, **zip** and **unzip:** Compressing and Archiving Together, **ls -l:** Listing File Attributes, The -d Option: Listing Directory Attributes, File Ownership, File Permissions, **chmod:** Changing File Permissions,

Directory Permissions, Changing File Ownership.

Unit3: **10 Hours**

vi Basics, Input Mode—Entering and Replacing Text, Saving Text and Quitting—The ex Mode, Navigation, Editing Text, Undoing Last Editing Instructions (U and U), Repeating the Last Command (.) , Searching for a Pattern (/ and ?), Substitution—Search and Replace (:s), The Shell's Interpretive Cycle, Shell Offerings, Pattern Matching—The Wild-card, Escaping and Quoting, Redirection: The Three Standard Files , /dev/null and /dev/tty: Two Special Files, Pipes, **tee:** Creating a Tee, Command Substitution, Shell Variables, Process Basics, **ps:** Process Status, System Processes (-e or -a), Mechanism of Process Creation, Internal and External Commands, Process States and Zombies, Running Jobs in Background, **nice:** Job Execution with Low Priority, Killing Processes with Signals, Job Control.

Unit4: **10 Hours**

The Shells, Environment Variables, The Common Environment Variables Aliases (**bash** and **ksh**), Command History (**bash** and **ksh**), In-line Command Editing (**bash** and **ksh**), Miscellaneous Features (**bash** and **ksh**), The Initialization Scripts, File Systems and Inodes, Hard Links, Symbolic Links and **ln,** The Directory, **umask:** Default File and Directory Permissions, Modification and Access Times, **find:** Locating Files, The Sample Database, **pr:** Paginating Files, **head:** Displaying the Beginning of a File, **tail:** Displaying the End of a File. **cut:** Slitting a File Vertically, **paste:** Pasting Files, **sort:** Ordering a File, **uniq:** Locate Repeated and Non repeated Lines, **tr:** Translating Characters, An Example: Displaying a Word-count List.

Unit5: **12 Hours**

grep: Searching for a Pattern, Basic Regular Expressions (BRE) —An Introduction, Extended Regular Expressions (ERE) and egrep, **sed:** The Stream Editor, Line Addressing, Using Multiple Instructions (-E and -F), Context Addressing ,Writing Selected Lines to a File (w), Text Editing, Substitution (s), Basic Regular Expressions Revisited, Shell Scripts, **read:** Making Scripts Interactive, Using Command Line Arguments, **exit** and Exit Status of Command, The Logical Operators && and ||—Conditional Execution,The **if** Conditional, Using **test** and [] to Evaluate Expressions, The **case** Conditional **expr:** Computation and String Handling, $0: Calling a Script by Different Names, **while:** Looping, **for:** Looping with a List,**set** and **shift:** Manipulating the Positional Parameters,The Here Document (<<) , **trap:** Interrupting a Program, Debugging Shell Scripts with set -x , Sample Validation and Data Entry Scripts , root: The System Administrator's Login, The Administrator's Privileges, Maintaining Security, User Management, Startup and Shutdown,Managing Disk Space , Device Files, Handling Floppy Diskettes, **cpio:** A Backup Program, **tar:** The "Tape" Archive Program.

TEXT BOOKS / REFERENCES:

Text Books:

1. Sumitabha Das: UNIX Concepts and Applications, 4th Edition, Tata McGraw Hill, 2006.

Reference Books:

1. Kenneth Roson et al: UNIX - The Complete Reference, Osborne/McGraw Hill, 2005.
2. Steve Montsugu: Using UNIX, Prentice Hall India, 2009.
3. M G Venkateshmurthy: UNIX and Shell Programming, Pearson Education, 2005.

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	

I	I	Theory	02	02	00	00	26

Course No	**Course Title**	**Pre Requisites**
MCA160	Soft Skill - I - Professional Communication	NIL

COURSE ASSESSMENT METHOD:

1. Internal Assessment 5 Events [3 Written Tests, 2 Events; Event-1: Presentation, Event-2: Group Discussion] Marks: 50 [10* 5 Events].
2. Semester End Exam [50 Marks, 1½ Hours].

COURSE OUTCOMES:

Upon successful completion of this course, the student will be able to:

1. Understand the basic techniques of communication. and barriers of communication and tips to overcome those barriers..
2. Understand importance of listening and develop good listening skills.
3. Implement different concepts, standards and preparation for an effective presentation.
4. Communicate technically and in groups effectively

TOPICS COVERED:

UNIT:1 - Basics of Technical Communication, Barriers to Communication **06 Hours**

Introduction, Process of Communication, Language as a Tool, Levels of Communication, Communication Networks, Importance of Technical Communication. Definition of Noise, Classification of Barriers.

UNIT: 2 - Technology in Communication **05 Hours**

Impact of Technology, Software for Creating Messages, Software for Writing Documents, Software for Presenting Documents, Transmitting Documents, Effective use of Available Technology,

UNIT: 3 - Active Listening. **05 Hours**

Introduction, Types of Listening, Traits of good Listener, Active versus passive listening, implications of effective listening.

UNIT: 4 - Effective Presentation Strategies. **05 Hours**

Introduction, Defining purpose, Analyzing Audience and Locale, Organizing Contents, preparing outline, Visual Aids, Understanding Nuances of Delivery, Kinesics, Proxemics, Paralinguistic's, Chronemics, Sample speech.

UNIT: 5 - Group Communication **05 Hours**

Introduction, Group Discussion, Organizational Group discussion, Group discussion as part of selection process

Meetings, conferences.

TEXT BOOKS/ REFERENCES:

1. Meenakshi Raman and Sangeeta Sharma: Technical Communication - Principles and Practices, Oxford University Press, 2010.
2. M.Ashraf Rizivi: Effective Technical Communication, Tata McGraw Hill, 2009.

ADDITIONAL LEARNING SOURCES:

1. https://www.ted.com/topics/communication
2. https://nptel.ac.in/courses/109104031/
3. www.directionservice.org/cadre/section4.cfm.

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
I	I	Laboratory	01	00	03	00	48

Course No	Course Title	Pre Requisites
MCA13L	Digital Electronics Lab	NIL

COURSE ASSESSMENT METHOD:

1. Internal Assessment [Test-1 + Test-2 + Observation book + Record + Viva : 15 + 15 + 05 + 05 +10] Total Marks: 50

COURSE OUTCOMES:

Upon successful completion of this course, the student will be able to:

1. Understand the need and importance of Lex and Yacc tools.
2. Understand and the usage of different regular expressions.
3. Understand the structure and working principle of Lex and Yacc programs.
4. Write, execute and debug the Lex programs.
5. Write, execute and debug the Yacc programs.

LIST OF EXPERIMENTS COVERED:

PART-A

Exercise on digital electronics:

1. Realization of all basic gates.
2. Realization of all universal gates.
3. Realization of X-OR and X-NOR operations using basic gates.
4. Realization of X-OR and X-NOR operations using universal gates.
5. Design of Half adder and Full adder using 2-input NAND gates.

6. Verify the operations of half adder using NAND gate.
7. Verify the operations of full adder using NAND gate.
8. Verify addition of two binary numbers using 4-bit adder chip.
9. Design and implement Excess-3 to BCD code converter using 4-bit adder chip.
10. Construct and implement the Boolean expression using AND, OR and NOT gates.
11. Construct and implement the Boolean expression using only NAND gates.
12. Verify and implement half and full subtract or using 2-input NAND gates.

PART-B

Exercises on Office Tools**:**

Note: Exercises on the following topics to be conducted using popular software packages available.

1. Word Processing

Creation of word document, formatting and editing text, inserting headers and footers, Use of Mail-merge facility

2. Spread-sheet

Building and editing spread-sheet, formatting spread-sheet, working with formulas and functions, sort or filter spread-sheet, working with charts, and automating spread-sheet tasks with macros

3. Computer Presentation

Learning commands, Preparation of slides, inserting texts, graphs, etc., color changing, automatic presentation of slides, changing time settings, object linking and embedding, etc.

TEXT BOOKS / REFERENCES:

1.Tokheim : Digital Electronics principles and applications, 6th edition, <cgraw hill , 2012.
2. Carl Hamacher, Z. varnesic and S.Zaky : Computer Organization, 5th edition, Mcgraw Hill, 2010
3. Leland L beck, D Manjula: System Software, 3rd edition, pearson education, 2010.

ADDITIONAL LEARNING SOURCES:

1. www.tutorialspoint.com/computer_logical_organization/.
2. www.freetechbooks.com/computer-organization-and-architecture-f56.html

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
I	I	Laboratory	01	00	03	00	48

Course No	Course Title	Pre Requisites
MCA14L	C programming lab	NIL

COURSE ASSESSMENT METHOD:

Internal Assessment [Test-1 + Test-2 + Observation book + Record + Viva : 15 + 15 + 05 + 05 +10] Marks: 50

COURSE OUTCOMES:

Upon successful completion of this course, the student will be :

1. Able to understand the basic terminology used in computer programming
2. Able to write, compile and debug programs in C language.
3. Able to use different data types in a computer program.
4. Able to design programs involving decision structures, loops and functions.
5. Able to understand the dynamics of memory by the use of pointers.

LIST OF EXPERIMENTS COVERED:

1. Calculate the salary of an employee given his basic pay, HRA = 10% of basic pay, TA=5% of his basic pay and deductions IT = 2.5% of his basic pay .

2. Check whether the given number is perfect number.

Definition: A perfect number is a positive integer that is equal to the sum of its proper positive divisors, that is, the sum of its positive divisors excluding the number itself. Example - The first perfect number is 6, because 1, 2, and 3 are its proper positive divisors, and 1 + 2 + 3 = 6.

3. Solve quadratic equations for the given values of a,b,c for all its instances.

4. Generate all Amstrong numbers upto n.

Definition: An Armstrong number of three digits is an integer such that the sum of the cubes of its digits is equal to the number itself. Example - 371 is an Armstrong number, since 3**3 + 7**3 + 1**3 = 371.

5. Write a menu driven C program to :
   a. Insert an element into an array
   b. Delete an element from the array (first occurrence) .

6. Write a Menu Driven C Program to
   a. Accept a string from the user
   b. Encode the string.
   c. Decode the string

Apply the following procedure to encode it.

1. Convert each character in a string to its ASCI value.
2. Add an integer value to it and display the encoded string
3. Decode the string using reverse procedure and display.

7. Write a C program to multiply two matrices that satisfy the constraint of matrix multiplication .

8. Write a C program to find the saddle point of a matrix.

Definition: Given a RxC Matrix, A, i.e. R rows and C columns we define a Saddle-Point as Saddle_Pt (A) for a row I and column j is that A(i,j) that is the minimum of Row i and the maximum of Col j.

9. Write a C program to implement a magic square of size n.

Definition: A magic square is an arrangement of numbers (usually integers) in a square grid, where the numbers in each row, and in each column, and the numbers that run diagonally in both directions, all add up to the same number.

10. Write a Menu driven C program to
    a. Accept two numbers n and m
    b. Sum of all integers ranging from n to m
    c. Sum of all odd integers ranging from n to m
    d. Sum of all even integers ranging from n to m

Note: Display an error message if n > m. Create functions for each of the options

11. Write a Menu Driven C Program to implement the following using recursion:
    a. Factorial of a number
    b. Fibonacci series
    c. Largest / Smallest of n numbers stored in an array
    d. Binary Search

12. Create a structure Complex Number having real and imaginary part as properties. Write functions to perform the arithmetic operations on two complex numbers.

13. Define a structure called student having the properties of student_id, student name and branch of the student with a sub structure of marks(3 subjects). Write a Menu Driven C Program to :
    a. Add new student detail
    b. Delete a student detail
    c. Display all student details
    d. Display the name of the student with the best / worst marks

14. Write a C Program to remove all white spaces and newline characters from a file.

15. Find whether a given word exists in the file. If it exists display the location of the word

16. Write a C program to copy one file content to another file without using inbuilt functions.

TEXT BOOKS / REFERENCES:

1. Programming in ANSI C, Third Edition, E.Balaguruswamy. 6th Edition (2013)
2. The C Programming Language, Brian W Kernighan, Dennis M Rtchie, PHI, 2nd Edition.
3. Let us C, 4th Edition, Yashwanthkanetkar.
4. Programming with C, Byron Gottfried, Tata McGraw-Hill edition.

ADDITIONAL LEARNING SOURCES:

1. http://www.worldbestlearningcenter.com/index_files/c_tutorial_lesson.htm
2. http://www.cluster2.hostgator.co.in/files/writeable/uploads/hostgator99706/file/letusc-yashwantkanetkar.pdf
3. http://fineprojects.blogspot.in/2012/03/let-us-c-by-yashwant-kanetkar-solved.html

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week	Total Hours/ Semester

				Theory	Laboratory	Tutorials	
I	II	Laboratory	01	00	03	00	48

Course No	Course Title	Pre Requisites
MCA15L	Linux Lab	NIL

COURSE ASSESSMENT METHOD:

Internal Assessment [Test-1 + Test-2 + Observation book + Record + Viva : 15 + 15 + 05 + 05 +10]
Marks: 50

COURSE OUTCOMES:

Upon successful completion of this course:

1. Able to understand the concepts of UNIX clearly.
2. Gets the practical exposure to the UNIX operating system.
3. Able to understand and see the working of the different commands of UNIX operating system.

LIST OF EXPERIMENTS COVERED:

1. a. Write a non-recursive shell script which accepts any number of arguments and prints them in the reverse order (For example, if the script is named rags, then executing args A B C should produce C B A on the standard output).
   b. 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.

2. a. Write a shell script that takes a valid directory name as an argument and recursively descend all the subdirectories, finds the maximum length of any file in that hierarchy and writes this maximum value to the standard output.
   b. Write a shell script that accepts a path name 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.

3. a. 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.
b. 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.

4. a. Create a script file called file-properties that reads a file name entered and outputs it properties.
b. Write a shell script that accept one or more filenames as argument and convert all of them to uppercase, provided they exist in current directory.

5. a. 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.

b. 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.

6. a. 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.

b. Write a shell script to find smallest of three numbers that are read from keyboard.

7. a. Write a shell script using expr command to read in a string and display a suitable message if it does not have at least 10 characters.

b. Write a shell script to compute the sum of number passed to it as argument on command line and display the result.

8. a. Write a shell script that compute gross salary of an employee, accordingly to rule given below.

If basic salary is < 15000 then HRA=10% of basic 7 DA=90% of basic.

If basic salary is >=15000 then HRA=500 of basic & DA=98% of basic.

b. Write a shell script that delete all lines containing a specific word in one or more file supplied as argument to it.

9. a. 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.

b. 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.

10. a. Write a shell script that determine the period for which a specified user is working on system.

b. 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.

11. a. Write a shell script that accepts two integers as its argument and compute the value of first number raised to the power of second number.

b. 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.

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

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

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

b. 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.

i. Electrical 34

ii. Mechanical 67
iii. Electrical 80
iv. Computer Science 43
v. Mechanical 65
vi. Civil 198
vii. Computer Science 64

14. 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.

II Semester

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
I	II	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA210	System Software	Computer Organization

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, Event-1 Quiz, Event-2. Test] Total Marks: 50 [10* 5].
2. Semester End Exam [Total Marks:100 Marks, Duration: 3 Hours]

COURSE OBJECTIVES:

The objective of this course is to teach the students:

1. Different parameters of machine architecture, relationship between system software and machine architecture.
2. Basics design principles and implementation of assemblers.
3. The role and working of linkers and loaders, the structure and working of editors and interactive debugging systems.
4. Working and design of macro processors.
5. The concept, structure and purpose of lexers and parsers.

COURSE OUTCOMES:

Upon successful completion of this course, the student will be able to:

1. Understand basics of system software, machine architecture, different data formats and addressing modes in SIC & SIC/XE.

2. Understand structure and working of assemblers, design principles.
3. Learn loader and linker functionalities, structure of editors and interactive debugging systems.
4. Understand the working, design and structures of a macro processor.
5. Learn fundamentals of scanner or Lexer and Parser and implement the lexer and parser using LEX and YACC tools.

TOPICS COVERED:

UNIT:1 - Machine Architecture **06 Hours**

Introduction, System Software and Machine Architecture, Simplified Instructional Computer (SIC) - SIC Machine Architecture, SIC/XE Machine Architecture, SIC Programming Examples.

UNIT:2 - Assemblers **12 Hours**

Basic Assembler Function - A Simple SIC Assembler, Assembler Algorithm and Data Structures, Machine Dependent Assembler Features - Instruction Formats & Addressing Modes, Program Relocation.

Machine independent Assembler Features – Literals, Symbol-Definition Statements, Expression, Program Blocks, Control Sections and Programming Linking, Assembler Design Operations - One-Pass Assembler, Multi-Pass Assembler, Implementation Examples - MASM Assembler.

UNIT:3 - Loaders, Linkers, Editors and Debugging Systems **14 Hours**

Basic Loader Functions - Design of an Absolute Loader, A Simple Bootstrap Loader, Machine-Dependent Loader Features – Relocation, Program Linking, Algorithm and Data Structures for a Linking Loader; Machine-Independent Loader Features - Automatic Library Search, Loader Options, Loader Design Options - Linkage Editor, Dynamic Linkage, Bootstrap Loaders, Implementation Examples - MS-DOS Linker.

Text Editors - Overview of Editing Process, User Interface, Editor Structure, Interactive Debugging Systems - Debugging Functions and Capabilities, Relationship With Other Parts Of The System, User-Interface Criteria.

UNIT:4 - Macro Processor **08 Hours**

Basic Macro Processor Functions - Macro Definitions and Expansion, Macro Processor Algorithm and Data Structures, Machine-Independent Macro Processor Features - Concatenation of Macro Parameters, Generation of Unique Labels, Conditional Macro Expansion, Keyword Macro Parameters, Macro Processor Design Options - Recursive Macro Expansion, General-Purpose Macro Processors, Macro Processing Within Language Translators, Implementation Examples - MASM Macro Processor, ANSI C Macro Processor.

UNIT:5 - Lex and Yacc **12 Hours**

Lex and Yacc - The Simplest Lex Program, Recognizing Words With LEX, Symbol Tables, Grammars, Parser-Lexer Communication, A YACC Parser, The Rules Section, Running LEX and YACC, Using LEX - Regular Expression, Examples of Regular Expressions.
Compilers- Basic compiler functions: rammers, lexical analysis, syntactic analysis, code generation. Design options, The Yacc Compiler-Compiler.Using YACC – Grammars, Recursive Rules, Shift/Reduce Parsing, What YACC Cannot Parse, A YACC Parser - The Definition Section, The Rules Section, Symbol Values and Actions.

TEXT BOOKS / REFERENCES:

1. Leland. L.Beck, D Manjula: System Software, 3rd Edition, Pearson Education, 2010.
2. John.R.Levine, Tony Mason and Doug Brown: Lex and Yacc, O'Reilly, SPD, 2009.
3. D.M.Dhamdhere: System Programming and Operating Systems, 2nd Edition, Tata McGraw - Hill, 2008.

ADDITIONAL LEARNING SOURCES:

1. http://techterms.com/definition/systemsoftware.
2. Studymca.blogspot.com/2009/01/system-software_24.html.

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
I	II	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA220	Operating System	Computer Organization and Architecture

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, Any 2 Events from the list [Test, Presentation, Quiz] Marks: 50 [10* 5 Events].
2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, the student will be able to:

Co1 Distinguish different styles of operating system design. Understand device and functions in operating systems as part of a uniform device abstraction.

Co2 Have an understanding of disk organization and file system structure. Be able to give the rationale for virtual memory abstractions in operating systems.

Co3 Understand the main principles and techniques used to implement Processes and threads as well as the different algorithms for process scheduling.

Co4 Understand the main mechanisms used for inter-process communication. Understand the main related to concurrency and different synchronization mechanisms.

Co5 Have the ability to evaluate security risks in operating systems and understand the role operating systems can and should play in establishing security.

TOPICS COVERED:

UNIT:1 Introduction to OS, System Structures **8 Hours**

What operating systems do, Computer System organization, computer system architecture, Operating System structure; Operating System operations, Process management ,Memory management; Storage management, Protection and security, Distributed system; Special-purpose systems Computing environments, Operating System Services, User - Operating System interface; System calls, Types of system calls; System programs Operating System design and implementation; Operating System structure; Virtual machines; Operating System generation; System boot.

UNIT:2 Process Management **10 Hours**

Process concept, Process Scheduling Operation on Processes Inter-process Communication Multi-Threaded Programming: Overview, Multithreading models Thread Libraries, Threading issues Process Scheduling: Basic concepts, Scheduling Criteria Scheduling Algorithms Scheduling Algorithms Multiple Processor Scheduling Thread Scheduling.

UNIT:3 - Process Synchronization And Deadlocks **12 Hours**

Synchronization: concepts, critical section problem, peterson's solution, synchronization hardware, semaphores, bounded buffer problem, readers-writers problem, dining philosophers problem. Deadlocks: System model; Deadlock Characterization Methods for handling deadlocks; Deadlock Prevention Deadlock Avoidance; Deadlock Avoidance ; Deadlock detection; Recovery from Deadlock.

UNIT:4 - Memory Management **10 Hours**

Memory Management strategies: Background, Swapping, Contiguous Memory Allocation Paging Structure of the page table, Segmentation Virtual Memory Management: Background, Demand Paging Copy on write, Page Replacement Page Replacement (continued) ,Allocation of Frames, Thrashing.

UNIT:5 - File System, Security And Protection **12 Hours**

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. Mass Storage Structures: Disk Structure, Disk Attachment Disk Scheduling , Disk Management, Swap space Management, Goals of Protection, Principles of protection, Domain of protection. Access Matrix, Implementation of Access Matrix. Access control, Revocation of Access rights, Capability-based system.

TEXT BOOKS / REFERENCES:

1. Operating System Principles: A.,Silberschatz, P. Galvin, G. Gagne, 9th Edition, Wiley – India, 2012.
2. Operating Systems-A Concept based Approach: D.M. Dhamdhere, 2nd Edition, Tata McGraw-Hill, 2002.
3. Introduction to Operating systems- Concepts and Practice: P.C.P Bhatt, 2nd Edition, PHI, 2008.

ADDITIONAL LEARNING SOURCES:

1. http://nptel.iitk.ac.in/

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
I	II	Theory	04	04	00	00	52

Course No	Course Title	Prerequisites
MCA230	Data Structures with C++	Problem Solving using C

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, Any 2 Events from the list [Test, Presentation, Quiz, Projects, assignments, Viva-Voce etc. Marks: 50 [10* 5 Events]

2. Semester End Exam [100 Marks, 3 Hours.

COURSE OUTCOMES:

Upon successful completion of this course, the student will be able to:
1. OOP concepts and C++ Programming concepts.
2. Understand Abstract data types such as arrays, structures, strings and polynomials.
3. Implement various ADTs such as STACK, QUEUE, LIST and Tree.
4. Implement various searching and sorting algorithms.
5. Able to implement efficient linear and non-linear data structures.

TOPICS COVERED:

UNIT I: Introduction and overview of C++ Programming 12 Hours

C++ Class Overview- Class Definition, OOPs concepts ,Objects, Class Members, Access Control, Class Scope, Inheritance and Polymorphism ,Constructors and destructors, parameter passing methods, Inline functions, static class members, this pointer, friend functions, dynamic memory allocation and deallocation (new and delete), exception handling.

UNIT:2 - Introduction to Data Structures 12 Hours

Information and its meaning: Abstract Data Types, Sequences as Value Definitions, ADT for Varying length character Strings, Data Types, Pointers and review of Pointers, Data Structures. Arrays : Array as an ADT, Using One-dimensional Arrays, Implementing One-Dimensional Arrays, Arrays as Parameters, String as an ADT.

UNIT 3: The Stack 12 Hours

Definition and examples, Primitive operations, Example, The stack as an ADT, Representing stacks, Implementing the pop, push operations using function overloading, Examples for infix , postfix, and prefix expressions, Basic definition and Examples. Applications of Stacks: Expression Evaluations, Expression conversion, Recursion as application of stack,Properties of recursive definition or algorithm. Binary search, Towers of Hanoi problem.

UNIT 4: Queues and Lists 10 Hours

The queue and its sequential representation, the queue as ADT, Basic operations using using polymorphism and

inheritance, Priority queue, Array implementation of a priority queue. Linked lists, inserting and removing nodes from a list, Linked implementations of stacks, Linked implementation of queues, linked list as a data Structure. Example of list operations.

UNIT 5: Linked Lists **8 Hours**

Other list structures: Circular lists, Stack as circular lists, doubly linked lists. Application of linked lists: Stacks, Queues, double-ended queues, priority queues. Sorting and Searching: Applications and implementation with function overloading . Bubble sort, Heap Sort, Merge Sort. Basic Search Techniques, Tree : Definition and representation, Types of trees ,Tree searching: Inserting into a Binary Search Tree, Deleting form a binary search tree.

TEXT BOOKS/ REFERENCES :

1. Herbert Scheldt: The Complete Reference C++, 6th Edition, Tata McGraw Hill 2013.
2.Data Structures Using C and C++ by Aaron.M.Tenenbaum, Yedidyah Langsam and Moshe J. Augenstein , PHI, Edition, 2011.
3. Data structures, Algorithms and Applications in C++, S. Sahni, University Press (India) Pvt.Ltd, 2nd Edition.

ADDITIONAL LEARNING SOURCES:

1. http://www.tutorialspoint.com/Data-Structures-in-C++-Online-Training/classid=13

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
I	II	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA240	Software Engineering-I	NIL

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, Any 2 Events from the list [Test, Presentation, Quiz. Projects, Group discussion, assignments, Viva-Voce etc.] Marks: 50 [10* 5 Events].

2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, the student will be able to:
1. An ability to identify, formulate, analyze, and solve problems, as well as identify the computing Requirements appropriate to their solutions.
2. An ability to design, implement, and evaluate software based systems, components, or programs of varying complexity that meet desired needs, satisfy realistic constraints, and demonstrate an accepted design and development principles.
3. An ability to apply knowledge of computing, mathematics, science, and engineering appropriate to the discipline.
4. An ability to use current techniques, skills, and tools necessary for professional practice
5.Modeling and design of software systems and in the analysis of tradeoffs inherent in design decisions.

TOPICS COVERED:

UNIT – 1 **11 Hours**

Introduction: professional software development, software engineering ethics, case studies,
Software Process: software process model, process activities, the rational unified process,
Agile Software Development: Agile methods, plan-Driven and agile-development, Extreme programming, agile project Management, Scaling Agile methods.

UNIT – 2 **10 Hours**

Requirements Engineering: Functional and Non-functional requirements; the software requirements document. Requirements Engineering Processes: Requirement specification, Requirements elicitation and analysis; Requirements validation; Requirements management.
System Modeling: Context models, Interaction models, structural models, Behavioral models; Model-driven engineering.

UNIT – 3 **11 Hours**

Architectural Design: Architectural Design decisions, architectural views, architectural patterns, Application architectures.
Design and Implementation: object-oriented design using the UML, Design patterns, Implementation issues, open source development.
Software Testing: Development Testing, Test-Driven testing, Release testing, user testing,

UNIT **10 Hours**

Software Evolution: Evolution process, program evolution dynamics, software maintenance,
Legacy system development.
Dependability and Security: Dependability properties, Availability and reliability, Safety, Security
Dependability and security specification: Risk- driven requirements specification, safety specification, Reliability specification, Security specification, Formal specification.

UNIT-5 **10 Hours**

Dependability Engineering: Redundancy and diversity, Dependable process, Dependable system architectures, Dependable programming.
Security Engineering: Security risk management, Design for security, System survivability.
Dependability and Security Assurance: Static analysis, reliability testing, Security testing, process assurance,

TEXT BOOKS / REFERENCES:

1. Ian Somerville: Software Engineering 9th Edition, Pearson Education, 2012
2. Roger.S.Pressman: Software Engineering-A Practitioners approach, 7th Edition, Tata McGraw Hill, 2007.
3. Pankaj Jalote: An Integrated Approach to Software Engineering, Wiley India, 2009.

ADDITIONAL LEARNING SOURCES:

1. https://www.iiitd.edu.in/~jalote/

2. Software Engineering Tutorial, www.tutorialspoint.com/software_engineering

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
I	II	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA250	Computer Networks-I	NIL

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, Any 2 Events from the list [Test, Quiz, Assignments etc.] Marks: 50 [10* 5 Events].

2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, the student will be able to:

1. Summarize the basic concepts, principles and techniques of data communications along with the layers of OSI and TCP/IP model.
2. The knowledge of how transmission medium in a network controlled by the physical layer that connects network components together.
3. Independently understand the concept of links, nodes and data transmission issues in the network.
4. Ability to understand and implement DLC protocols either connectionless or connection-oriented.
5. Capability to understand wired and wireless Ethernet along with IPv4 addressing and performance of the network-layer.

TOPICS COVERED:

Unit – 1: Basics of Data Communications **8 Hours**

Data Communications: Components, Data Representation, Data Flow, Networks; Network Criteria, Physical Structures, Network Types: LAN, WAN, Switching, The Internet, Accessing the Internet, The Internet History: Early History, Birth of the Internet, Internet Standards, Network Models: Protocol Layering: Principles of Protocol Layering, Logical Connections, TCP/IP Protocol Suite: Layered Architecture, Layers in the TCP/IP Protocol Suite, Encapsulation and Decapsulation, Addressing, Multiplexing and Demultiplexing, The OSI Model; OSI versus TCP/IP, Lack of OSI Model's Success.

UNIT – 2: Physical Layer **12 Hours**

Introduction to Physical Layer, Data and Signals, Periodic Analog Signals, Digital Signals, Transmission Impairment, Data Rate Limits, Performance, Digital-to-Digital Conversion, Analog-to-Digital Conversion, Transmission Modes, Digital-to-Analog Conversion, Analog-to-Analog Conversion, Switching: Circuit-Switched Networks, Packet Switching, Datagram Networks, Virtual Circuit Networks.

Unit – 3: Data Link Layer - 1 **10 Hours**

Introduction to Data-Link Layer, Link-Layer Addressing: Address Resolution Protocol (ARP), Error Detection and Correction: Introduction, Types of Errors, Redundancy, Detection versus Correction, Coding, Block

coding: Error Detection, Cyclic Code: Cyclic Redundancy Check, Polynomials, Cyclic Code Analysis and its Advantages, Checksum, Forward Error Correction: Using Hamming Distance, Using XOR, Chunk Interleaving.

Unit – 4: Data Link Layer - 2 **10 Hours**

Data Link Control (DLC), DLC Services, Data-Link Layer Protocols, Simple Protocol, Stop-and-Wait Protocol, Piggybacking, HDLC: Configurations and Transfer Modes, Framing, Point-To-Point Protocol (PPP): Services, Framing, Transition Phases, Multiplexing, Random Access: ALOHA, CSMA, CSMA/CD, CSMA/CA, Controlled Access: Reservation, Polling, Token Passing, Channelization: FDMA, TDMA, CDMA.

Unit – 5: LANs and Network Layer **12 Hours**

Ethernet Protocol, Standard Ethernet: Characteristics, Addressing, Access Method, Efficiency of Standard Ethernet, Implementation, Changes in the Standard, Fast Ethernet (100 MBPS): Access Method, Physical Layer, 10 Gigabit Ethernet, IEEE 802.11 Project, Bluetooth, Introduction to Network Layer, Network-Layer Services: Packetizing, Routing and Forwarding, Packet Switching: Datagram Approach, Virtual-Circuit Approach, Network Layer Performance: delay Throughput, Packet loss, Congestion Control, IPv4 addresses, IPv6 addresses, Forwarding of IP Packets.

TEXT BOOKS / REFERENCES:

1. B. A. Forouzan, Data Communications and Networking, 5th Edition, McGraw Hill Education (India) Private Limited, 2013.

1. William Stallings, Data and Computer Communications, 10th Edition, Pearson, 2013.
2. Larry L. Peterson and Bruce S. David: Computer Networks – A Systems Approach, 5th Edition, Elsevier, 2011.
3. Andrew S. Tanenbaum, Computer Networks, Fourth Edition, PHI, 2008.
4. Fred Halsall, Data Communications, Computer Networks and Open Systems, 4th Edition, Pearson Education, 2005.
5. www.nptel.ac.in/courses

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
I	II	Theory	02	02	00	00	26

Course No	Course Title	Prerequisites
MCA260	Soft Skills –II (Ethics in IT)	NIL

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, Any 2 Events from the list [Test, Presentation, Quiz, Projects, assignments, Viva-Voce etc.] Marks: 50 [10* 5 Events].

2. Semester End Exam [50 Marks, 2 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, the student will be able to:
1. Understanding do's and don'ts in information technologies industry.
2. Maintaining privacy in software development .
3. Maintaining good relation between employer and employees.
4. Ethics while using workers like contingent and contractors
5. Knowing employee rights.

TOPICS COVERED:

Unit-1: An Overview of Ethics. **4 Hours**
What are Ethics : Definition of ethics, Importance of Integrity, Ethics in the Business World, Ethics in Information Technology (IT).

UNIT:2 Ethics for IT Professionals and IT users **4 Hours**
IT professionals ,The Ethical behavior of IT professionals, IT Users : Common ethical issues for IT users, Supporting the ethical practices of IT users.

UNIT:3 Privacy **6 Hours**
Privacy Protection and the Law : The right of privacy and recent history of privacy protection, Key Privacy and Anonymity Issues.

Unit 4 : Intellectual Property **6 Hours**
What is Intellectual property: copy rights, patents ,trade secret laws, Key Intellectual property issues : plagiarism, reverse engineering , Open source code , competitive intelligence, cyber squatting.

UNIT:5 - Software Development; Employer/Employee Issues , **6 Hours**

Strategies to Engineer Quality Software, Key Issues in software development. Use of Non-traditional workers, Whistle Blowing. Case Studies and Discussion.

TEXT BOOKS/ REFERENCES:

1. George Reynolds: Ethics in Information Technology, 2nd Edition, Thomson Course Technology, 2007.(Chapters: 1, 2, 4, 7, 8).
2. Mike W. Martin, Roland Schinzinger, Ethics in Engineering Tata McGraw Hill Edition 2007.

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
I	II	Laboratory	01	00	03	00	48

Course No	Course Title	Pre Requisites
MCA21L	System Software Laboratory	Unix & C- Programming

COURSE ASSESSMENT METHOD:

1. Internal Assessment [Test-1 + Test-2 + Observation book + Record + Viva : 15 + 15 + 05 + 05 +10] Total Marks: 50

COURSE OBJECTIVES:

The objective of this course is to teach the students:

1. The need and importance of Lex and Yacc tools.
2. Various regular expressions and their appropriate usage.
3. Structure of a Lex and Yacc program.
4. To write, analyze and run lex programs
5. To write, analyze and run Yacc programs

COURSE OUTCOMES:

Upon successful completion of this course, the student will be able to:

1. Understand the need and importance of Lex and Yacc tools.
2. Understand and the usage of different regular expressions.
3. Understand the structure and working principle of Lex and Yacc programs.
4. Write, execute and debug the Lex programs.
5. Write, execute and debug the Yacc programs.

LIST OF EXPERIMENTS COVERED:

Lex Programs

1. To count the number of vowels and consonants in a given string.
2. To count the number of characters, words, spaces and lines in a given input file.
3. To count the number of positive and negative integers and positive and negative fractions.
4. To check whether the given sentence is a simple or compound sentence.
5. To count the number of strings, commands and drives.
6. To count the number of lines in a given c program, also eliminate them and copy the program into a s separate file.
7. To count the number of ‘scanf’ and ‘printf’ statements in a c program and replace them with ‘readf’ and ‘writef’ statements.
8. To recognize a valid arithmetic expression and identify the identifiers and operators present, print them separately.

9. To recognize and count the number of identifiers in a given input file.
10. To read data from a input file and count the number of characters, spaces, words and lines, write the output to a separate file.
11. To read data from a input file and count the number of characters, spaces, words and lines, write the output to a separate file using command line arguments.
12. To count number of vowels, consonants, digits and spaces in a given input string.
13. To count the number of chars, spaces, digits and lines from a given data from the command line.
14. To count the number of comment lines from a C++ program file and remove the comments and write the output a new file.
15. To check the entered day is a week day or week end.
16. To check the given date falls under which quarter of the year.

Yacc Programs

To test the validity of a simple expression involving operators +,-,*,/.
To recognize nested IF control statements and display the number of levels of nesting.
To test the valid arithmetic expression that uses operators +,-,* and /.
To recognize a valid variable, which starts with a letter, followed by any number of letters or digits
To evaluate an arithmetic expression involving operators +,-,* and /:
To recognize strings 'aaab', 'abbb', 'ab' and 'a' using the grammar (a^nb^n, n>=0)
To recognize the grammar (a^nb, n>=10)
To demonstrate what Yacc cannot parse

TEXT BOOKS / REFERENCES:

4. John.R.Levine, Tony Mason and Doug Brown: Lex and Yacc, O'Reilly, SPD, 1999.
5. John.R.Levine: Flex and Bison, O'Reilly, SPD, 2012

ADDITIONAL LEARNING SOURCES:

1. http://dinosaur.compilertools.net/

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
I	II	Laboratory	01	00	03	00	48

Course No	Course Title	Prerequisites
MCA23L	Data Structures with C++ Lab	Problem Solving using C

COURSE ASSESSMENT METHOD:

Internal Assessment [Test-1+Test-2+Observation book + Record + Viva : 15 + 15 + 05 + 05 +10=50marks]

COURSE OUTCOMES:

Upon successful completion of this course, the student will be able to:

1.Able to implement oops concepts like.,class and object etc.,
2. Able to understand the importance of structure and abstract data type, and their basic usability in different Applications through different programming languages.
3. Able to analyze and differentiate different algorithms based on their time complexity.
4. Able to understand the linked implementation, and its uses both in linear and non-linear data structure.
5. Able to understand various data structure such as stacks, queues, trees, etc. to solve various computing problems.

LIST OF EXPERIMENTS COVERED:

PROGRAMS ON CLASS & OBJECT CREATION

1. Given that an EMPLOYEE class contains the following members: Data Members: Employee_Number, Employee_Name, Basic, DA, IT, Net_Sal Member Functions: to read data, to calculate Net_Sal and to print data members. Use constructor and destructor for Employee object.
Write a C++ program to read data on N employees and compute the Net_Sal of each employee (DA = 52% of Basic and Income Tax = 30% of the gross salary).
Perform the following functions:
a. Insert N employees information
b. Search for a particular employee details
c. Edit the information after search.
d. Swap two employees information
e. Display the list of all the employees

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

3.Write a C++ program to create a class called EXPRESSION. Using appropriate member functions convert a given valid Infix expression into postfix form. Display the infix and postfix expressions.

4. Write a C++ program to perform basic operations using function overloading on rational numbers using structures.

5. Write a C++ program to perform basic operations using polymorphism on 2 or more complex numbers. [Complex number is an ADT represented in the form of a+ib]

STACK
6. Write a C++ program to perform the basic operations on a stack of book information (MAX=5). Implement constructor and destructor for Bookinformation consists of ISBN, Number of pages, Author, and Price. Use function overloading for the basic operations.
7. Write a C++ program to evaluate the validity of an expression
8. Write a C++ program to evaluate a postfix expression.
9. Write a C++ program to convert an expression from infix to postfix.
REMARK: here top1 moves from left to right 1.e, with -1 as initialization and top2 moves from right to left with n as initialization
let the program have appropriate message in view of remarking all possible conditions.
Implement the above program for a structure called furniture using constructor and destrucor with details model

no, type name, and price let the array get generated dynamically
10. Write a C++ program to implement multiple stack of integers.
NOTE: Use Two dimentional array to represent the stack and additional of one dimentional array to store the top pointer values correspondingly. Use function and operator overlaoding for basic operations.

QUEUES

11. Write a C++ program to perform basic operations on queue of integers using function overloading, the program should provide the appropriate message to handle all concerned conditions
12. Write a C++program to perform basic operations on list of students information stored in circular queue. Let student information include regno, course title, year of study
13. Write a C++program to implement dual queue.

LINKED LIST

14. Write a C++program to implement stack using linked list.
15. Write a C++program to implement queue of structure pen using linked list.
16. Write a C++program to create the students mark list based on the rank. Let the student record contain student-id, name, total marks. Use function and operator overloading for display the contesnt of the list.
17. Write a C++program to perform operations using function overloading on a list of integers.
a. Creation of list using constructors.
b. Insertion of new element [At Front, from rear, based on the position]
c. Deletion of a node [At Front, from rear, based on the position]
d. Display the list using operator overloading.
e. Replace the content of one element by another element.
f. Swap two nodes
18. Write a C++program to perform the following operations on doubly linked list.
a. Creation of list by :
Insertion [At beginning, At end, In between]
Deletion [At beginning, At end, In between]
b. Display all the nodes.
c. Swap two nodes based on specific criteria.

TREES

19. Write a C++program to perform / implement the binary tree using array and hence perform the following
a. To print the left and right child of specified node
b. To print all the ancestors of a specified node
c. To print all the node in a specific level
d. To print only the leaf node
20. Write a C++program to perform / implement the binary tree using linked list and hence perform the following
a. To print the left and right child of specified node
b. To print all the ancestors of a specified node
c. To print all the node in a specific level
d. To print only the leaf node
21. Write a C++program with recursive routines to traverse the binary tree in all possible orders
a. Create a tree
b. Pre-Order traversal
c. In-Order traversal
d. Post-Order traversal
22. Write a C++program to construct a heap of n integers and hence sort them using heap sort algorithm
23. Implement the search techniques
a. Linear Search
b. Binary Search

TEXT BOOKS/ REFERENCES :

1. Herbert Scheldt: The Complete Reference C++, 6th Edition, Tata McGraw Hill 2013.

2.Data Structures Using C and C++ by Aaron.M.Tenenbaum, Yedidyah Langsam and Moshe J. Augenstein , PHI, Edition, 2011.
3. Data structures, Algorithms and Applications in C++, S. Sahni, University Press (India) Pvt.Ltd, 2nd Edition.

ADDITIONAL LEARNING SOURCES:

1. http://www.tutorialspoint.com/Data-Structures-in-C++-Online-Training/classid=13

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
I	II	Laboratory	01	00	01	00	48

Course No	Course Title	Pre Requisites
MCA25L	Computer Networks Lab-I	NIL

COURSE ASSESSMENT METHOD:

Internal Assessment [5 Events: 3 Practical Tests, Any 2 Events from the list [Record, Observation, Attendance, Viva etc.] Marks: 50 [10* 5 Events].

COURSE OUTCOMES:

Upon successful completion of this course, the student will be able to:

1. Ability to simulate and analyze computer networks with network simulator tool.
2. Hands-on experience on how transmission medium in a network, controlled by the physical layer that connects network components together.
3. The practical knowledge of links, nodes and data transmission issues in the network.
4. Simulate the techniques of data communications along with the layers of OSI and TCP/IP model.
5. Ability to understand Internet architecture.

LIST OF EXPERIMENTS

1. Study of different physical equipment used for networking.
2. Study of OSI reference model and TCP/IP reference model.
3. Understand the connection between two pc using peer to peer communication.
4. Study of Network topologies with advantages & disadvantages.
5. Study of basic network commands: ipconfig, hostname, ping <ip_address>, tracert <ip_address>, netstat<ip_address> etc.

6. Study of multiplexing and demultiplexing.
7. Conversion of Analog to Digital Signal using network simulator software tool.
8. Implement the data link layer framing methods such as character count, character stuffing and bit stuffing.
9. Implement the error detecting techniques:

- Parity check
- Cyclic Redundancy Check (CRC)
- Check Sum

10. Implement Error correcting Techniques
11. Implement Dijkstra 's algorithm to compute the Shortest path through a graph.
12. Study of IEEE 802.11 wireless standard.

TEXT BOOKS / REFERENCES:

1. B. A. Forouzan, Data Communications and Networking, 4th Edition, McGraw Hill, 2010.

1. William Stallings, Data and Computer Communications, 10th Edition, Pearson, 2013.
2. Larry L. Peterson and Bruce S. David: Computer Networks – A Systems Approach, 5th Edition, Elsevier, 2011.
3. Andrew S. Tanenbaum, Computer Networks, Fourth Edition, PHI, 2008.
4. Fred Halsall, Data Communications, Computer Networks and Open Systems, 4th Edition, Pearson Education, 2005.
5. www.nptel.ac.in/courses

III Semester

Department of Master of Computer Applications [MCA], SJCE, Mysuru							
Course Year	**Course Semester**	**Course Type**	**Credits**	**Contact Hours/ Week**			**Total Hours/ Semester**
				Theory	**Laboratory**	**Tutorials**	
II	**III**	**Theory**	**04**	**04**	**00**	**00**	**52**
Course No		**Course Title**				**Pre Requisites**	
MCA310		**Relational Database Management System**				**Data Structures**	

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2 Events] Marks: 50 [10* 5 Events]
2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, students will be

CO 1: Able to understand the modern data management issues, advantages of RDBMS over conventional file handling

CO 2: Analyse the concepts of data models and modeling notations, use of SQL, algebraic expressions/ operations on relational database.

CO 3: Handle the RDB, extraction / evaluation of DB using SQL using tuple relations and calculus.

CO 4: Exposed to entity-relation model, design of schema.

CO 5: Justify the need of normalization, normalize the RDB up to BCNF

TOPICS COVERED:

UNIT:1 – Introduction 10 Hours

Database-System Applications, Purpose of Database Systems, View of Data, Database Languages, Relational Databases, Database Design, Data Storage and Querying, Transaction Management,Dtabase Architecture, Data Mining and Information Retrieval, Specialty Databases, Database Users and Administrators, History of Database Systems.

Introduction to the Relational Model:

Structure of Relational Databases, Database Schema, Keys, Schema Diagrams, Relatinal Query Languages, Relational Operations.

UNIT:2 – Introduction to SQL 10 Hours

Overview of the SQL Query Language, SQL Data Definition, Basic Structure of SQL Queries, Additional Basic Operations, Set Operations, Null Values, Aggregate Functions, Nested Subqueries, Modification of the Database.

Intermediate SQL:

Join Expressions, Views, Transactions, Integrity Constraints, SQL Data Types and Schemas, Authorization.

UNIT:3 - Advanced SQL 10 Hours

Accessing SQL From a Programming Language, Functions and Procedures, Triggers, Recursive Queries, Advanced Aggregation Features, OLAP.

Formal Relational Query Languages:

The Relational Algebra, The Tuple Relational Calculus, The Domain Relational Calculus.

UNIT:4-Database Design and the E-R Model 10 Hours

Overview of the Design Process, The Entity-Relationship Model, Constraints, Removing Redundant Attributes in Entity Sets, Entity-Relationship Diagrams, Reduction to Relational Schemas, Entity-Relationship Design Issues, Extended E-R Features, Alternative Notations for Modeling Data, Other Aspects of Database Design.

UNIT:5 – Relational Database Design 12 Hours

Features of Good Relational Design, Atomic Domains and First Normal Form, Decomposition Using Functional Dependencies, Functional Dependency Theory, Algorithm for Decomposition, Decomposition Using Multivalued Dependencies, More Normal Forms, Database-Design Process, Modeling Temporal Data.

TEXT BOOKS / REFERENCES:

Text Books :

1. "Database System Concepts", A.Silberschatz, Henry.F.Korth, S.Sudharshan, 6th Edition.

2. "Database Systems", Ramez Elmasri, Shamkant.B.Navathe, 6th Edition.

Reference Books:

1. "Database Management Systems", Raghu Ramakrishnan and J Gehrke 3rd Edition
2. "An Introduction to Database System" C.J.Date , AKannan, S..Swamynathan 8th Edition

ADDITIONAL LEARNING SOURCES:

1. www.tutorialspoint.com/sql/sql-rdbms-concepts.htm
2. nptel.ac.in/courses/106106093/6
3. msdis.missouri.edu/resources/gis_advanced/pdf/relational.pdf

CO-PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
C	H	H	H	H	M	L	K	L	M	M	M	M
C	H	H	M	H	M	L	M	L	M	M	M	M
C	H	H	H	H	M	L	M	L	L	M	L	L
C	H	M	M	M	L	L	L	L	L	M	L	L
C	H	L	M	H	L	L	M	M	M	L	M	L

Department of Master of Computer Applications [MCA], SJCE, Mysuru

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
II	III	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA320	Java Programming	C and C++ Concepts

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2 Events] Marks: 50 [10* 5 Events]
2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, students will be able to,

CO 1: Understand the basics and features of Java.

CO 2: Understand the principles of object oriented programming, packages and interfaces in Java.

CO 3: Implement Java program for multithread, synchronization concepts and able to implement exception handling in programs for handling errors.

CO 4: Demonstrate string and event handling and learn to use simple data structures like arrays and members of classes found in the Java API.

CO 5: Use graphical user interface using applets, swing components and networking concepts in Java.

TOPICS COVERED:

UNIT:1 – Java Basics **10 Hours**

History of Java, Java buzzwords, data types, variables, scope and life time of variables, arrays, operators, expressions, control statements, type conversion and casting, simple java program, concepts of classes, objects, constructors, methods, access control, this keyword, garbage collection, overloading methods and constructors, parameter passing, recursion, nested and inner classes, exploring string class

UNIT:2 – OOP concepts in Java, Packages and Interfaces **10 Hours**

Hierarchical abstractions, Base class object, subclass, subtype, substitutability, forms of inheritance- specialization, specification, construction, extension, limitation, combination, benefits of inheritance, costs of inheritance. Member access rules, super uses, using final with inheritance, polymorphism- method overriding, abstract classes, the Object class.

Defining, Creating and Accessing a Package, Understanding CLASSPATH, importing packages, differences between classes and interfaces, defining an interface, implementing interface, applying interfaces, variables in interface and extending interfaces. Exploring java.io.

UNIT:3 - Exception handling & Multi Threading **10 Hours**

Concepts of exception handling, benefits of exception handling, Termination or presumptive models, exception hierarchy, usage of try, catch, throw, throws and finally, built in exceptions, creating own exception sub classes. String handling, Exploring java.util.
Differences between multi threading and multitasking, thread life cycle, creating threads, thread priorities, synchronizing threads; inter thread communication, thread groups, daemon threads. Enumerations, auto boxing, annotations, generics.

UNIT:4 - String and Event Handling **10 Hours**

String fundamentals, String Constructors, Three string related language features, The Length() method, Obtaining the characters within the string, String comparison, using indexOf() and lastindexOf(), changing the case of the characters within the string, String buffer and String builder.
Events, Event sources, Event classes, Event Listeners, Delegation event model, handling mouse and keyboard events, Adapter classes. The AWT class hierarchy, user interface components- labels, button, canvas, scrollbars,

text components, check box, check box groups, choices, lists panels – scroll pane, dialogs, menu bar, graphics, layout manager – layout manager types – border, grid, flow, card and grid bag.

UNIT:5 – Applets, Swings and Networking with Java.Net **12 Hours**

Concepts of Applets, differences between applets and applications, life cycle of an applet, types of applets, creating applets, passing parameters to applets.
Swings- Introduction, limitations of AWT, MVC architecture, components, containers, exploring swing- JApplet, JFrame and JComponent, Icons and Labels, text fields, buttons – The JButton class, Check boxes, Radio buttons, Combo boxes, Tabbed Panes, Scroll Panes, Trees, and Tables.
Networking fundamentals, Networking Classes & Interfaces, The InetAddress class, The Socket class, URL class, URL connection class, Http URL connection class, Exploring collection frame work, Collection overview, Collection classes and interfaces, Array class.

TEXT BOOKS / REFERENCES:

Text books:

1. Herbert Schildt. Java - The Complete Reference, Ninth Edition. Oracle Press, McGraw Hill Education (India) Edition- 2014.

Reference books:

1. Cay S. Horstmann, Gary Cornell. Core Java, Core Java Volume-1 – Fundamentals, 9th edition, Pearson Education, 2014.
2. Timothy Budd, "Understanding Object-oriented programming with Java", Updated Edition, Pearson Education, 2000.

ADDITIONAL LEARNING SOURCES:

1. http://www.oracle.com/technetwork/java/index-jsp-135888.html
2. http://www.javaworld.com/article/2074929/core-java
3. http://www.javaworld.com/
4. http://www.learnjavaonline.org/
5. https://www.codecademy.com/learn/learn-java
6. http://www.tutorialspoint.com/java/
7. http://www.java-examples.com/
8. http://www.homeandlearn.co.uk/java/java.html

CO - PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
CO	H	H	H	H	M	M	M	M	H	M	M	M
CO	H	M	M	H	M	M	H	M	M	M	L	L
CO	H	M	M	H	M	H	H	M	H	M	L	L
CO	H	M	M	H	M	M	M	M	M	M	L	L
CO	H	H	H	H	M	M	H	M	M	M	L	L

Department of Master of Computer Applications [MCA], SJCE, Mysuru

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
II	III	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA330	Software Engineering-II	Software Engineering-I

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2 Events] Marks: 50 [10* 5 Events]
2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, students will be

- **CO 1:** Show the importance of systems analysis and design in solving complex problems.
- **CO 2:** An ability to perform analysis, design and implementation of a proposed software system based on the analysis techniques using advanced static/dynamic UML models.
- **CO 3:** An ability to analyze and develop domain and application of software.
- **CO 4:** An ability to translate UML models into code using an OO programming language and understanding the related OOD techniques such as design by contract, refactoring, and test driven design as well as an ability to understand new/existing OO frameworks.
- **CO 5:** The ability to work in one or more significant application domains and to manage the development of software system.

TOPICS COVERED:

UNIT:1- Introduction to Modeling Concepts **10 Hours**

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 and Advanced 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-array associations; Aggregation; Abstract classes; Multiple inheritance; Metadata; Reification; Constraints; Derived data; Packages; Practical tips

UNIT:2 -State Modeling and Advanced State Modeling **10 Hours**

State Modeling: Events, States, Transitions and Conditions; State diagrams; State diagram behavior; Practical tips. 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 and Advanced Interaction Modeling

Interaction Modeling: Use case models; Sequence models; Activity models. Use case relationships; Procedural sequence models; Special constructs for activity models.

UNIT:3 -Process Overview, System Conception **10 Hours**

Process Overview: Development stages; Development life cycle. System Conception: Devising a system concept; elaborating a concept; preparing a problem statement.

Domain Analysis and Application Analysis

Overview of analysis; Domain class model; Domain state model; Domain interaction model; Iterating the analysis. Application Analysis: Application interaction model; Application class model; Application state model; adding operations.

UNIT:4-System Design and Class Design **12 Hours**

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. 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.

UNIT:5 -Implementation Modeling: **10 Hours**

Fine-tuning Classes, Fine-tuning Generalizations, Realizing Associations, Testing.

Software reuse: The reuse landscape, Application frameworks, software product lines, COTS product reuse, Distributed Software engineering: Distributed Software issues, client-server computing, architectural patterns for distributed systems, software as a service, Quality management: software quality, software standards, reviews and inspections, software Measurement and metrics.

TEXT BOOKS / REFERENCES:

Text books:

1. Michael Blaha, James Rumbaugh: Object-Oriented Modeling and Design with UML, 2nd Edition, Pearson Education / PHI, 2005. (Chapters 1 to 15)
2. Ian Sommerville : Software Engineering, 9th edition, Person Education Ltd, 2011

Reference books:

1. Grady Booch et al: Object-Oriented Analysis and Design with Applications, 3rd Edition,Pearson, 2007.
2. Mark Priestley: Practical Object-Oriented Design with UML, 2nd Edition, Tata McGraw-Hill, 2003.
3. K. Barclay, J. Savage: Object-Oriented Design with UML and JAVA, Elsevier, 2008.
4. Booch, G., Rumbaugh, J., and Jacobson, I.: The Unified Modeling Language User Guide, 2nd Edition, Pearson, 2005.
5. Simon Bennett, Steve McRobb and Ray Farmer: Object-Oriented Systems Analysis and Design Using UML, 2nd Edition, Tata McGraw-Hill, 2002.
6. Rumbaugh, Blaha, Premerhani, Eddy, Lorensen; Object Oriented Modeling and Design, PHI Latest Edition

ADDITIONAL LEARNING SOURCES:

1. https://nscnetwork.files.wordpress.com/2011/09/object-oriented-modeling-and-design.pdf
2. https://www-public.tem-tsp.eu/~gibson/Teaching/CSC7322/L7-UML.pdf

CO-PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
CO	H	H	H	M	M	M	H	H	H	M	H	H
CO	H	H	H	H	H	M	M	M	H	M	H	H
CO	H	H	H	M	H	M	H	H	H	M	H	H
CO	H	H	H	H	H	M	H	H	H	M	H	H
CO	H	H	H	H	H	H	H	H	H	M	H	H

Department of Master of Computer Applications [MCA], SJCE, Mysuru

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
II	III	Theory	05	04	00	01	52

Course No	Course Title	Pre Requisites
MCA 340	Analysis and Design of Algorithms	--

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2 Events] Marks: 50 [10* 5 Events]
2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, students will be

CO 1: Understanding basic ideas about algorithms.

CO 2: Understanding the concepts of time and space complexity, worst case, average case and best case complexities and the big-O notation.

CO 3: Understanding the range of behaviors of algorithms and the notion of tractable and intractable problems.

CO 4: Knowing and understanding a wide range of searching and sorting algorithms.

CO 5: Developing efficient algorithms for simple computational tasks and computing complexity measures of algorithms, including recursive algorithms using recurrence relations.

TOPICS COVERED:

UNIT:1 – Introduction and the fundamentals of the Analysis of Algorithm Efficiency 12 Hours

Notion of Algorithm, Fundamentals of Algorithmic Problem Solving, Important Problem Types, Fundamental data Structures. Analysis Framework, Asymptotic Notations and Basic efficiency classes, Mathematical analysis of Recursive and Non-recursive algorithms, Examples

UNIT:2 - Brute Force and Divide and Conquer **10 Hours**

Selection Sort and Bubble Sort, Sequential Search and String Matching, Exhaustive Search, Merge-sort, Quick-sort, Binary Search, Binary tree Traversals and related properties.

UNIT: 3 - Decrease-and-Conquer, Transform-and-Conquer **10 Hours**

Insertion Sort, Depth First search and Breadth First Search, Topological sorting, Algorithms for Generating Combinatorial Objects. Presorting, Balanced Search Trees, Heaps and Heap sort, Problem Reduction.

UNIT 4 - Space and Time Tradeoffs and Dynamic Programming: **10 Hours**

Sorting by Counting, Input Enhancement in String Matching, Computing a binomial coefficient, Warshall's and Floyd's Algorithms, The Knapsack Problem and Memory Functions,

UNIT 5 Greedy Technique and Limitations of Algorithm Power: **10 Hours**

Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm, Huffman Trees, Lower-Bound Arguments, Decision Trees, **P**, **NP** and **NP**-Complete Problems.

Text Book:

1. Anany Levitin: Introduction to the Design and Analysis of Algorithms, Pearson Education, 2003.

References:

1. Coremen T.H., Leiserson C.E., and Rivest R.L.: Introduction to Algorithms, PHI, 1998.
2. Horowitz E., Sahani S., Rajasekharan S.: Computer Algorithms, Galgotia Publications, 2001.

CO-PO MAPPING:

	PO	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO**	**PO1**	**PO1**	**PO12**
CO	H	H	H	M	M	L	M	M	L	L	M	L
CO	H	M	L	M	M	L	L	L	L	M	M	M
CO	H	H	H	H	M	L	M	M	L	L	L	M
CO	M	M	L	M	L	L	L	L	M	L	M	L
CO	H	H	H	H	M	M	M	M	L	L	L	L

Department of Master of Computer Applications [MCA], SJCE, Mysuru

Course Semester	**Course Type**	**Credits**	**Contact Hours/ Week**	
			Theory	**Laboratory**
III	Theory	05	04	00

Course No	**Course Title**
MCA350	Computer Networks-II

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2 Events] Marks: 50 [10* 5 Events]

2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, students will be

CO 1: Understand the services of TCP and UDP.
CO 2: Ability to understand the basic concepts and techniques of Client-Server Application and Network Management.
CO 3: The knowledge of WWW, HTTP, and FTP Standards
CO 4: Independently understand the issues of Multimedia.
CO 5: Capability to understand the aspects of Network Security and Internet Security.

TOPICS COVERED:

UNIT:1 – Transport Layer **10 Hours**

Introduction to Transport-Layer: Transport-Layer Services; Transport-Layer Protocols: Port Numbers, User Datagram Protocol: User Datagram, UDP Services, UDP Applications, Transmission Control Protocol: TCP Services, TCP Features, Segment, A TCP Connection, State Transition Diagram, Windows in TCP, Flow Control, Error Control, TCP Congestion Control, TCP Timers.

UNIT:2 – Application Layer and Network Management **11 Hours**

Introduction to Application Layer, Services, Application-Layer Paradigms, Client-Server Programming: Application Programming Interface, Using Services of the Transport Layer, Iterative Communication using UDP, Iterative Communication using TCP, Concurrent Communication, Iterative Programming in C: General Issues, Iterative Programming using UDP, Iterative Programming using TCP, Network Management: Configuration Management, Fault Management, Performance Management, Security Management, Accuonting Management, SNMP: Managers and Agents, Management Components, An Overview, SMI, MIB, SNMP.

UNIT:3 - Standard Client-Server Protocols **10 Hours**

World Wide Web and HTTP: FTP: Two Connections, Control Connection, Data Connection, Security for FTP, E-Mail: Architecture, Web-Based Mail, TELNET: Local versus Remote Logging, Secure Shell (SSH): Components, Applications, Domain Name System (DNS): Name Space, DNS in the Internet, Resolution, Caching, Resource Records, DNS Messages, Registrars, DDNS, Security of DNS.

UNIT:4 – Multimedia **10 Hours**

Compression: Lossless Compression, Lossy Compression, Multimedia Data: Text, Image, Video, Audio, Multimedia in the Internet: Streaming Stored Audio/Video, Streaming Live Audio/Video, Real-Time Interactive Audio/Video, Real-Time Interactive Protocols: Rationale for new Protocols, RTP, RTCP, Session Intialization Protocol (SIP), H.323.

UNIT:5 – Network Security and Internet Security **11 Hours**

Security Goals, Attacks, Services and Techniques, Confidentiality: Symmetric-Key Ciphers, Asymmetric-Key Ciphers, Internet Security: Network-Layer Security: Two Modes, Two Security Protocols, Services Provided by IPSec, Security Association, Internet Key Exchange (IKE), Virtual Private Network (VPN), Transport-

Layer Security: SSL Architecture, Four Protocols, Application-Layer Security: E-mail Security, Pretty Good Privacy (PGP), S/MIME, Firewalls: Packet-Filter Firewall, Proxy Firewall.

TEXT BOOKS / REFERENCES:

Text books:

1. B. A. Forouzan, Data Communications and Networking, 5th Edition, McGraw Hill Education (India) Private Limited, 2013.

Reference books:

1. Alberto Leon-Garcia and Indra Widjaja: Communication Networks –Fundamental Concepts and Key architectures, 2nd Edition Tata McGraw-Hill, 2004.
2. Computer & Communication Networks, Nadir F Mir, Pearson Education, India.
3. Larry L. Peterson and Bruce S. David: Computer Networks – A Systems Approach, 5th Edition, Elsevier, 2011.
4. Andrew S. Tanenbaum, Computer Networks, Fourth Edition, PHI, 2008.
5. Fred Halsall, Data Communications, Computer Networks and Open Systems, 4th Edition, Pearson Education, 2005.

ADDITIONAL LEARNING SOURCES:

1. www.nptel.ac.in/courses
2. http://freevideolectures.com/Course/2276/Computer-Networks

CO-PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
CO	**L**	**M**	**L**	**H**	**M**	**L**	**L**	**L**	**M**	**L**	**L**	**L**
CO	**L**	**H**	**H**	**M**	**H**	**M**	**M**	**L**	**M**	**L**	**L**	**L**
CO	**L**	**H**	**H**	**M**	**H**	**M**	**M**	**L**	**M**	**L**	**L**	**L**
CO	**H**	**H**	**H**	**H**	**H**	**M**	**H**	**L**	**M**	**M**	**M**	**L**
CO	**H**	**H**	**H**	**H**	**H**	**H**	**H**	**L**	**M**	**L**	**M**	**L**

Department of Master of Computer Applications [MCA], SJCE, Mysuru

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
II	III	Theory	02	02	00	00	26

Course No	Course Title	Pre Requisites
MCA360	Soft Skills-III : Personal Effectiveness and Leadership	Soft Skills-I (Professional Communication)

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2-Events. Test] Total Marks: 50 [10* 5].
2. 2. Semester End Exam [Total Marks:50 Marks, Duration: 1.5 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, students will be

CO 1: Understand the strengths and weaknesses as individual, as a member of a group/organization using personality typing.

CO 2: Develop self awareness and to understand the concepts of Neuro-lingistic programming (NLP).

CO 3: Euip with skill and competence to apply positive psychology principles in a range of environments to increase individual and collective well being.

CO 4: Understand the concept of leadership, types, and theories and conceive the leadership potential in themselves.

CO 5: Develop knowledge, skills and understanding the leadership behavior and motivation concepts.

TOPICS COVERED:

UNIT:1 – Learning and Time Management **5 Hours**

Introduction, learning shock, overcoming barriers to learning, how do I learn? How do we learn?, learning styles and orientations, metaphors for learning, learning from experience: informal and self-directed learning. Introduction, what is time management? , preparation for change, the consequences, the good habits of time management, time management scenarios.

UNIT:2 – Creativity, Ideas Generation, group dynamics and team working **5 Hours**

Introduction, blocks in creative thinking, cycles and funnels of creativity, developing a whole brain approach to creativity, creative techniques:1, 2, 3 & 4.Team working: Introduction, features of successful team, group development, meetings, team roles.

UNIT:3 - Assertiveness and Negotiation skills **5 Hours**

Introduction, status work, rights of work, challenging situations, tricky feelings and emotions, cultural differences. Negotiation skills: Introduction, principles of negotiation, negotiating approaches, negotiation strategy and tactics, the negotiation process.

UNIT:4 – Individuals as Leaders **3 Hours**

Leadership is everyone’s business, leadership managerial roles, levels of analysis of leadership theory, leadership theory paradigms.

UNIT:5 – Leadership Traits, Ethics, Behavior and Motivation **8 Hours**

Personality traits and leadership, traits of effective leaders, leadership attitudes, ethical leadership. Leadership behavior and styles, leadership grid, major motivational theories, content motivation theories, process motivation theories, reinforcement theory.

TEXT BOOKS / REFERENCES:

Text books:

1. Lussier/Achua : Effective Leadership, 3rd Edition, Thomson Education , India edition, 2010.
2. Diana Winstantley : Personal Effectiveness , 1st Edition, Excel books, 2009.

Reference books:

1. Northouse, P. G. (2013). (6th ed.) Leadership: Theory and Practice. Thousand Oaks:CA: Sage Publications.

ADDITIONAL LEARNING SOURCES:

1. https://www.ted.com/topics/leadership/personal effectiveness.
2. http://www.emeraldinsight.com.

CO-PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
CO	L	L	L	L	L	H	M	H	H	L	H	M
CO	L	L	L	M	M	H	M	M	H	M	M	M
CO	L	L	L	L	L	M	H	H	H	H	H	M
CO	L	L	L	L	L	M	M	H	H	H	M	M
CO	L	L	L	L	L	M	M	H	M	M	H	H

Department of Master of Computer Applications [MCA], SJCE, Mysuru

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
II	III	Laboratory	01	00	03	00	48

Course No	Course Title	Pre Requisites
MCA31L	Relational Database Management System Laboratory	Data Structures

COURSE ASSESSMENT METHOD:

1. Internal Assessment [Test-1+ Test-2 + Observation book + Record + Viva: 15 + 15 + 05 + 05 +10] Marks: 50.

COURSE OUTCOMES:

Upon successful completion of this course, students will be

CO 1: Analyse the concepts of data models and modeling notations
CO 2: Use of SQL, algebraic expressions/ operations on relational database
CO 3: To introduce the concepts of transactions and transaction processing
CO 4: Design the queries to handle all the basic and advanced operations on RDBMS
CO 5: Design of applications which handles the operations involved in RDBMS

List of Programs:

1. Consider the following relations:
Student (snum: integer, sname: string, major: string, level: string, age: integer)
Class (name: string, meets at: string, room: string, d: integer)
Enrolled (snum: integer, cname: string)
Faculty (fid: integer, fname: string, deptid: integer)
The meaning of these relations is straightforward; for example, Enrolled has one record per student-class pair such that the student is enrolled in the class. Level is a two character code with 4 different values (example: Junior: JR etc)
Write the following queries in SQL. No duplicates should be printed in any of the answers.
 Find the names of all Juniors (level = JR) who are enrolled in a class taught by Prof. XYZ
 Find the names of all classes that either meet in room R128 or have five or more Students enrolled.
 Find the names of all students who are enrolled in two classes that meet at the same time.
 Find the names of faculty members who teach in every room in which some class is taught.
 Find the names of faculty members for whom the combined enrollment of the courses that they teach is less than five.

2. The following relations keep track of airline flight information:
Flights (no: integer, from: string, to: string, distance: integer, Departs: time, arrives: time, price: real)
Aircraft (aid: integer, aname: string, cruisingrange: integer)
Certified (eid: integer, aid: integer)
Employees (eid: integer, ename: string, salary: integer)
Note that the Employees relation describes pilots and other kinds of employees as well; certified for some aircraft, and only pilots are certified to fly.

Write each of the following queries in SQL.
 Find the names of aircraft such that all pilots certified
 Find the names of aircraft such that all pilots certified to operate have salaries more than Rs.80, 000.
 For each pilot who is certified for more than three aircrafts, find the eidand the maximum cruisingrangeof the aircraft for which she or he is certified
. Find the names of pilots whose salary is less than the price of the cheapest route from Bengaluru to Frankfurt.
For all aircraft with cruisingrangeover 1000 Kms, .find the name of the aircraft and the average salaryof all pilots certified for this aircraft.
 Find the names of pilots certified for some Boeing aircraft.
 Find the aids of all aircraft that can be used on routes from Bengaluru to New Delhi.

3. Consider the following database of student enrollment in courses & books adopted for each course.
STUDENT (regno: string, name: string, major: string, bdate:date)
COURSE (course #:int, cname:string, dept:string)
ENROLL (regno:string, course#:int, sem:int, marks:int)

BOOK _ ADOPTION (course# :int, sem:int, book-ISBN:int)
TEXT (book-ISBN:int, book-title:string, publisher:string, author:string)

 Create the above tables by properly specifying the primary keys and the foreign keys.
 Enter at least five tuples for each relation.
 Demonstrate how you add a new text book to the database and make this book be adopted by some Department.
 Produce a list of text books (include Course #, Book-ISBN, Book-title) in the alphabetical order for Courses offered by the 'CS' department that use more than two books.
 List any department that has all its adopted books published by a specific publisher.
 Generate suitable reports.
 Create suitable front end for querying and displaying the results.

4. The following tables are maintained by a book dealer.
AUTHOR (author-id:int, name:string, city:string, country:string)
PUBLISHER (publisher-id:int, name:string, city:string, country:string)
CATALOG (book-id:int, title:string, author-id:int, publisher-id:int, category-id:int, year:int, price:int)
CATEGORY (category-id:int, description:string)
ORDER-DETAILS (order-no:int,book-id:int, quantity:int)
 Create the above tables by properly specifying the primary keys and the foreign keys.
 Enter at least five tuples for each relation.
 Give the details of the authors who have 2 or more books in the catalog and the price of the books is greater than the average price of the books in the catalog and the year of publication is after 2000.
 Find the author of the book which has maximum sales.
 Demonstrate how you increase the price of books published by a specific publisher by 10%.
 Generate suitable reports.
 Create suitable front end for querying and displaying the results.

5. Consider the following database for a banking enterprise
BRANCH(branch-name:string, branch-city:string, assets:real)
ACCOUNT(accno:int, branch-name:string, balance:real)
DEPOSITOR(customer-name:string, accno:int)
CUSTOMER(customer-name:string, customer-street:string, customer-city:s
LOAN(loan-number:int, branch-name:string, amount:real)
BORROWER(customer-name:string, loan-number:int)
 Create the above tables by properly specifying the primary keys and the foreign keys
 Enter at least five tuples for each relation
 Find all the customers who have at least two accounts at the Main branch.
 Find all the customers who have an account at all the branches located in a specific city.
 Demonstrate how you delete all account tuples at every branch located in a specific city.
 Generate suitable reports also suitable front end for querying and displaying the results.

TEXT BOOKS / REFERECES

Text Books :

1. "Database System Concepts", A.Silberschatz, Henry.F.Korth, S.Sudharshan, 6th Edition.
2. "Database Systems", Ramez Elmasri, Shamkant.B.Navathe, 6th Edition.

Reference Books:

1. "Database Management Systems", Raghu Ramakrishnan and J Gehrke 3rd Edition
2. "An Introduction to Database System" C.J.Date , AKannan, S..Swamynathan 8th Edition

ADDITIONAL LEARNING SOURCES:

1. www.tutorialspoint.com/sql/sql-rdbms-concepts.htm
2. nptel.ac.in/courses/106106093/6
3. msdis.missouri.edu/resources/gis_advanced/pdf/relational.pdf

CO-PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
CO	H	H	H	H	M	L	K	L	M	M	M	M
CO	H	H	M	H	M	L	M	L	M	M	M	M
CO	H	H	H	H	M	L	M	L	L	M	L	L
CO	H	M	M	M	L	L	L	L	L	M	L	L
CO	H	L	M	H	L	L	M	M	M	L	M	L

Department of Master of Computer Applications [MCA], SJCE, Mysuru

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
II	III	Laboratory	01	00	03	00	48

Course No	Course Title	Pre Requisites
MCA32L	Java Programming Laboratory	C and C++ Concepts

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 2 Written Tests, Viva-voce + Record + Observation] Marks: 50 [15+15+10+5+5]

COURSE OUTCOMES:

Upon successful completion of this course, students will be able to,

CO 1: Implement basic Java programs

CO 2: Implement and use principles of object oriented concepts, packages and interfaces in Java.

CO 3: Implement Java program for multithread, synchronization, exception handling concepts.

CO 4: Demonstrate string, event handling, use simple data structures like arrays and members of classes found in the Java API.

CO 5: Demonstrate graphical user interface using applets, swing components and networking concepts in Java.

LIST OF PROGRAMS TO BE COVERED:

1. Write a program to display Hello world
2. Write a program to ODD and EVEN numbers
3. Write a program to find factorial of number
4. Write a program to find the sum of the digits of a given number
5. Write a program to swap two numbers without using a temporary variable
6. Write a program accepts a name and display the name with greeting message using Class.
7. Write a program to generate a salary for an employee using class, object, constructors, methods and access control. Different parameters to be considered are Emp_No, Emp_Name, Age, Basic, DA, HRA, CA, PT, IT.
8. Write a program to generate a sales report for a sales executive using class, object, constructors, methods and access control. Different parameters to be considered are Emp_No, Emp_Name, Sales_Q1, Sales_Q2, Sales_Q3, Sales_Q4.
9. Write a JAVA Program to demonstrate Constructor Overloading and Method Overloading.
10. Write a JAVA Program to implement Inner class and demonstrate its Access protection.
11. Write a program in Java for String handling which performs the following:
    a. Checks the capacity of StringBuffer objects.
    b. Reverses the contents of a string given on console and converts the resultant string in upper case.
    c. Reads a string from console and appends it to the resultant string of ii.
12. Write a JAVA Program to demonstrate Inheritance.
13. Simple Program on Java for the implementation of Multiple inheritance using
    a. interfaces to calculate the area of a rectangle and triangle.
14. Write a JAVA program which has
    a. A Class called Account that creates account with 500Rs minimum balance, a deposit() method to deposit amount, a withdraw() method to withdraw amount and also throws LessBalanceException if an account holder tries to withdraw money which makes the balance become less than 500Rs.
    b. A Class called LessBalanceException which returns the statement that says withdraw amount (Rs) is not valid.
    c. A Class which creates 2 accounts, both account deposit money and one account tries to withdraw more money which generates a LessBalanceException take appropriate action for the same.
15. Write a JAVA program using Synchronized Threads, which demonstrates Producer
16. Consumer concept.
17. Write a JAVA program to implement a Queue using user defined Exception Handling
(also make use of throw, throws.).
18. Complete the following:
    a. Create a package named shape.
    b. Create some classes in the package representing some common shapes like Square,
    c. Triangle and Circle. Import and compile these classes in other program.
19. Write a JAVA Program
    a. Create an enumeration Day of Week with seven values SUNDAY through SATURDAY. Add a method is Workday( ) to the DayofWeek class that returns true if the value on which it is called is MONDAY through FRIDAY. For example, the call DayOfWeek.SUNDAY.isWorkDay ( )

returns false.

20. Write a JAVA program which has
    a. A Interface class for Stack Operations
    b. A Class that implements the Stack Interface and creates a fixed length Stack.
    c. A Class that implements the Stack Interface and creates a Dynamic length Stack.
    d. A Class that uses both the above Stacks through Interface reference and does the Stack operations that demonstrates the runtime binding.
21. Write a JAVA program to print a chessboard pattern.
22. Write a JAVA Program which uses FileInputStream / FileOutPutStream Classes.
23. Write JAVA programs which demonstrate utilities of LinkedList Class.
24. Write a JAVA program which uses Datagram Socket for Client Server Communication.
25. Write a JAVA applet program, which handles keyboard event.
26. Write a JAVA Swing program, to design a form.

TEXT BOOKS / REFERENCES:

Text books:

1. Herbert Schildt. Java - The Complete Reference, Ninth Edition. Oracle Press, McGraw Hill Education (India) Edition- 2014.

Reference books:

1. Cay S. Horstmann, Gary Cornell. Core Java, Core Java Volume-1 – Fundamentals, 9th edition, Pearson Education, 2014.

2. Timothy Budd, "Understanding Object-oriented programming with Java", Updated Edition, Pearson Education, 2000.

ADDITIONAL LEARNING SOURCES:

1. http://www.oracle.com/technetwork/java/index-jsp-135888.html
2. http://www.javaworld.com/article/2074929/core-java
3. http://www.javaworld.com/
4. http://www.learnjavaonline.org/
5. https://www.codecademy.com/learn/learn-java
6. http://www.tutorialspoint.com/java/
7. http://www.java-examples.com/
8. http://www.homeandlearn.co.uk/java/java.html

CO - PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
CO	H	L	L	H	M	M	H	M	M	M	M	M
CO	H	M	H	H	L	M	H	M	M	M	H	M
CO	H	M	H	H	M	M	H	M	M	M	H	M
CO	H	L	M	H	L	M	H	M	M	M	M	M
CO	H	M	M	H	M	M	H	M	M	M	H	M

Department of Master of Computer Applications [MCA], SJCE, Mysuru							
Course Year	**Course Semester**	**Course Type**	**Credits**	**Contact Hours/ Week**			**Total Hours/ Semester**
				Theory	**Laboratory**	**Tutorials**	
II	III	Laboratory	01	00	03	00	48

Course No	**Course Title**	**Pre Requisites**
MCA34L	Software Engineering Laboratory	Knowledge object-oriented concepts

COURSE ASSESSMENT METHOD:

1. Internal Assessment [Test-1 + Test-2 + Observation book + Record + Viva : 15 + 15 + 05 + 05 +10] Marks: 50

COURSE OUTCOMES:

Upon successful completion of this course, students will be

CO 1: Show the importance of systems analysis and design in solving complex problems.

CO 2: An ability to perform analysis, design and implementation of a proposed software system based on the analysis techniques using advanced static/dynamic UML models.

CO 3: An ability to analyze and develop domain and application of software.

CO 4: An ability to translate UML models into code using an OO programming language and understanding the related OOD techniques such as design by contract, refactoring, and test driven design as well as an ability to understand new/existing OO frameworks.

CO 5: The ability to work in one or more significant application domains and to manage the development of software system.

LIST OF EXPERIMENTS COVERED:

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. For Analysis and Design models, diagrams such as Use-case, Class Diagram, Sequence/Collaboration Diagram Should be drawn with suitable scenario, activity diagram, component diagram & deployment diagram. Generation of Test cases and utilization of test cases.

1. Passport automation system.
2. Book bank
3. Exam Registration
4. Stock maintenance system.
5. Online course reservation system
6. E-ticketing
7. Software personnel management system

8. Credit card processing
9. E-book management system
10. Recruitment system
11. Foreign trading system
12. Conference Management System
13.BPO Management System

TEXT BOOKS / REFERENCES:

Text books:

1. Michael Blaha, James Rumbaugh: Object-Oriented Modeling and Design with UML, 2nd Edition, Pearson Education / PHI, 2005. (Chapters 1 to 15)
2. Grady Booch et al: Object-Oriented Analysis and Design with Applications.

Reference books:

1. Mark Priestley: Practical Object-Oriented Design with UML, 2nd Edition, Tata McGraw-Hill, 2003.
2. K. Barclay, J. Savage: Object-Oriented Design with UML and JAVA, Elsevier, 2008.
3. Booch, G., Rumbaugh, J., and Jacobson, I.: The Unified Modeling Language User Guide, 2nd Edition,Pearson, 2005.
5. Simon Bennett, Steve McRobb and Ray Farmer: Object-Oriented Systems Analysis and Design Using UML, 2nd Edition, Tata McGraw-Hill, 2002

ADDITIONAL LEARNING SOURCES:

1. https://nscnetwork.files.wordpress.com/2011/09/object-oriented-modeling-and-design.pdf
2. https://www-public.tem-tsp.eu/~gibson/Teaching/CSC7322/L7-UML.pdf

CO-PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	M	M	M	H	H	H	M	H	H
CO2	H	H	H	H	H	M	M	M	H	M	H	H
CO3	H	H	H	M	H	M	H	H	H	M	H	H
CO4	H	H	H	H	H	M	H	H	H	M	H	H
CO5	H	H	H	H	H	H	H	H	H	M	H	H

IV Semester

Department of Master of Computer Applications [MCA], SJCE, Mysuru							
Course Year	**Course Semester**	**Course Type**	**Credits**	**Contact Hours/ Week**			**Total Hours/ Semester**
				Theory	**Laboratory**	**Tutorials**	
II	**IV**	**Theory**	**04**	**04**	**00**	**00**	**52**

Course No	**Course Title**	**Pre Requisites**
MCA410	**Web Programming**	**Unix, Computer concepts**

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2 Events] Marks: 50 [10* 5 Events]
2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, students will be

CO 1: Understand and apply basic design and usability principals when creating content for the internet.

CO 2: Demonstrate practical skills in workflows and methods used to create websites within a professional environment.

CO 3: Make informed and critical decisions regarding client development using XHTML and JavaScript.

CO 4: Design and implement reasonably sophisticated server-side applications using one or more suitable technologies.

CO 5: Have the knowledge to critically analyze and evaluate web applications. Construct websites using a variety of skills and techniques.

TOPICS COVERED:

UNIT:1 - Fundamentals of Web, XHTML, CSS **10 Hours**

Internet, WWW, Web Browsers and Web Servers, URLs, MIME, HTTP, Security, The Web Programmers Toolbox. XHTML: Basic syntax, Standard structure, Basic text markup, Images, Hypertext Links. Lists, Tables, Forms, Frames.CSS: Introduction, Levels of style sheets, Style specification formats, Selector forms, Property value forms, Font properties, List properties, Color, Alignment of text, The box model, Background images, The and tags, Conflict resolution.

UNIT:2 – JavaScript **12 Hours**

Overview of JavaScript, Object orientation and Javascript, 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, Examples.

JavaScript and HTML Documents, Dynamic Documents with Javascript.The JavaScript execution environment, The Document Object Model, Element access in Javascript, Events and event handling, Handling events from the Body elements, Button elements, Text box and Password elements, The DOM 2 event model, The navigator object, DOM tree traversal and modification. Introduction to dynamic documents, 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.

UNIT:3 - XML **10 Hours**

Introduction, Syntax, Document structure, Document type definitions, Namespaces, XML schemas, Displaying raw XML documents, Displaying XML documents with CSS, XSLT style sheets, XML processors, Web services. Perl, 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.

UNIT:4 – CGI,PHP **10 Hours**

The Common Gateway Interface; CGI linkage; Query string format; CGI.pm module; A survey example; Cookies 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, Cookies, Session tracking.

UNIT:5 - Ruby, Rails **10 Hours**

Origins and uses of Ruby, Scalar types and their operations, Simple input and output, Control statements, Arrays, Hashes, Methods, Classes, Code blocks and iterators, Pattern matching. Overview of Rails, Document requests, Processing forms, Rails applications with Databases, Layouts.

TEXT BOOKS / REFERENCES:

Text books:

1. Robert W. Sebesta: Programming the World Wide Web, 8th Edition, Pearson education, 2014.

2. Chris Bates: Web Programming Building Internet Applications, 3rd Edition, Wiley India, 2007**.**

Reference books:

1.M. Deitel, P.J. Deitel, A. B. Goldberg: Internet & World Wide Web How to Program, 3rd Edition, Pearson education, 2004

ADDITIONAL LEARNING SOURCES:

1. https://www.cis.upenn.edu/~bcpierce/courses/629/lectures.html
2. https://www.webstepbook.com/supplements.shtml

CO-PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
CO	H	M	M	H	H	L	M	M	M	L	M	L
CO	H	M	M	H	H	L	M	M	M	L	M	L
CO	H	M	M	H	H	L	M	M	M	L	M	L
CO	H	M	M	H	H	L	M	M	M	L	M	L
CO	H	M	M	H	H	L	M	M	M	L	M	L

Department of Master of Computer Applications, SJCE, Mysuru

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
II	IV	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA420	Advanced Java Programming	Java Programming

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2 Events] Marks: 50 [10* 5 Events]
2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, students will be

CO 1: Understanding Enterprise Application Architecture and J2EE, Learning and developing and Database Interaction Techniques using JDBC.
CO 2: Learning the concept of Server side applications using Servlets and developing the server side applications using Servlets.
CO 3: Able to understand JSP and write applications using JSP
CO 4: Able to design and developing components using Java Beans and Annotations
CO 5: Apply the concept of middle & Data Based application layers for an enterprise application using Enterprise Java Beans (EJB) and learn to develop the Server Side Component Models.

TOPICS COVERED:

UNIT:1 – J2EE Basics, Java Database Connectivity (JDBC) 08 Hours

Introduction to J2EE, Different architectures for application development, advantages of multi tier architecture for over other architectures, Different components and technologies of J2EE.

Introduction on JDBC, Talking to database, Essential JDBC program, Using Prepared Statement Object, Interactive SQL tool, JDBC in action result sets, Batch Updates, Mapping, Basic JDBC data types and Advanced data types.

UNIT:2 – Server Side Scripting using Servlets 08 Hours

CGI Technologies, Limitations, Advantages of Servlet over CGI , Servlet Structure, Servlet packaging, HTML building utilities, Lifecycle, Single Thread model 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.

UNIT:3 - Java Server Pages (JSP) 12 Hours

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 scriplets, for example using scriplets 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 auto flush attributes, Info attribute, errorpage and iserrorpage attributes, isThreadsafe attribute, extends attribute, language attribute, Including files and applets in JSP pages, Using java beans component in JSP documents.

UNIT:4 – Java Annotations and Java Beans 08 Hours

Creating packages, interfaces, JAR files and annotations, The core java API package, java.lang sub package, built-in annotations. 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.

UNIT:5 - Enterprise Java Beans (EJB) & Server Side Component Model 16 Hours

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**.**

Introduction on EJB components, Session Beans, 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 / REFERENCES:

Text Books:

1. Marty Hall, Larry Brown. Core Servlets and Java Server Pages. Volume 1: Core Technologies. Second Edition. (Chapter 3,4,5,6,7,8,9,10,11,12,13,14).

2. Java 6 Programming Black Book, Dreamtech Press. 2012 (Chapter 17,18,19,20,21,22,27,28,29,30).

3. Andrew 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, Eight Edition. Comprehensive coverage of the Java Language. Tata McGraw-Hill Edition – 2011.

ADDITIONAL LEARNING SOURCES:

1. www.j2eetutorials.50webs.com
2. www.docs.oracle.com/javaee/6/tutorial/doc/
3. www.tutorialspoint.com/listtutorials/java/j2ee/1
4. www.coderanch.com/t/534906/EJB-JEE/java/Tutorial-EE-Beginners
5. www.j2eebrain.com/
6. http://w3schools.invisionzone.com/index.php?showforum=52
7. http://www.javagenious.com/search/jsp-tutorial-w3schools/
8. http://gsraj.tripod.com/java/java_tutorial.html

CO-PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
CO	M	H	H	H	H	H	H	M	M	M	H	M
CO	H	M	H	M	M	M	H	L	M	M	M	L
CO	H	M	H	M	M	M	H	L	M	M	M	L
CO	H	M	M	M	M	M	H	L	M	M	M	L
CO	M	H	H	H	H	H	H	M	H	M	H	H

Department of Master of Computer Applications [MCA], SJCE, Mysuru							
Course Year	**Course Semester**	**Course Type**	**Credits**	**Contact Hours/ Week**			**Total Hours/ Semester**
				Theory	**Laboratory**	**Tutorials**	
II	**IV**	**Theory**	**05**	**04**	**00**	**01**	**52**
Course No		**Course Title**				**Pre Requisites**	
MCA430		**Cloud Computing**				**Computer Networks, Data Communication**	

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2 Events] Marks: 50 [10* 5 Events]
2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, students will be able to:

CO 1: Understand the architecture of Cloud Computing.

CO 2: Understand and use the service models and Deployment.

CO 3: Work on any real cloud service.

CO 4: Understand the service Management and Security of Cloud.

CO 5: Understand the Computing Paradigms and Cloud Computing.

TOPICS COVERED:

UNIT :1- Defining Cloud Computing , Assessing the Value Proposition, Understanding Cloud Architecture, Understanding Cloud Architecture 11 Hours

Defining Cloud Computing, Cloud types, The NIST model, The Cloud Cube Model, Deployment models, Service models, Examining the Characteristics of Cloud Computing, paradigm shift.Benefits of cloud computing, Disadvantages of cloud computing, Assessing the role of open standards.

Assessing the Value Proposition: Measuring the Cloud's Value, Early adopters and new application, The laws of cloudonomics, Cloud computing obstacles, Behavioral factors relating to cloud adoption, Measuring cloud computing costs, Avoiding Capital Expenditures, Right-sizing, Computing the total cost of ownership, Specifying service level agreements, Defining licensing models.

Understanding Cloud Architecture: Exploring the cloud computing stack, Composability, Infrastructure, Platforms. Virtual Appliances, Communication Protocols, Applications, Connecting to the Cloud, The Jolicloud Netbook OS, Chromium OS: The browser as an Operating System**.**

UNIT:2 - Understanding Service and Application by Type, Understanding Abstraction and Virtualization **10 Hours**

Understanding Service and Application by Type: Defining Infrastructure as a service (IaaS), Defining Platform as a Service (PaaS), Defining Software as a Service (SaaS), Defining Identity as a Service (IDaaS), and Defining Compliance as a Service (CaaS).

Understanding Abstraction and Virtualization: Using Virtualization technologies, Load Balancing and Virtualization, Understanding Hypervisors, Understanding Machine Imaging, Porting Applications. Capacity Planning, Defining Baseline and Metrics, Network Capacity, Scaling.

UNIT:3 - Exploring Platform as a Service, Understanding Cloud Security **11 Hours**

Exploring Platform as a Service: Defining Services, Using PaaS Application Frameworks, Managing the Cloud, Administrating the Clouds, Cloud Management Products, Emerging Cloud Management Standards.

Understanding Cloud Security: Securing the cloud, Securing data, Establishing identity and Presence.

UNIT:4 - Moving Application to the Cloud , Working with cloud-based storage, Using Webmail Services **10 Hours**

Moving Application to the Cloud: Application in the Cloud, Applications and cloud APIs.

Working with cloud-based storage: Measuring the digital universe, Provisioning Cloud Storage, Exploring Cloud Backup Solutions, Cloud Storage Interoperability.

Using Webmail Services: Exploring the cloud Mail Services, Working with syndication services.

UNIT: 5 - Communicating with the cloud , Working with Mobile Devices , Working with Mobile web Services **10 Hours**

Communicating with the cloud: Exploring instant messaging, Exploring collaboration technologies, Using social networks.

Working with Mobile Devices: Defining the Mobile Market, Using Smart phones with the Cloud.Working with Mobile web Services: Understanding Service Types, Mobile interoperability,performing Service Discovery, context-aware services, MEMS, Location awareness, push services, The BlackBerry push services, using SMS,Defining SMS,WAP and other Protocols.

TEXT BOOKS / REFERENCES:

Text Books:

1.Barrie Sosinsky “Cloud Computing Bible” Published by Wiley.

Reference Books:

1. Cloud Computing Principles and Paradigms by Rajkumar Buyya.

2. Cloud Computing Theory and Practice by Dan C.Marinescu.

ADDITIONAL LEARNING SOURCES:

1. www.scoopworld.in
2. www.cloudschool.com
3. www.techno-pulse.com

CO-PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	P10	P11	PO12
CO	H	M	H	M	L	H	M	M	M	H	M	L
CO	M	H	M	M	H	M	M	L	H	H	M	H
CO	H	M	L	H	M	H	H	H	L	M	M	M
CO	M	H	M	M	M	M	M	L	H	L	H	M
CO	H	M	H	M	L	M	L	H	M	M	H	L

Department of Master of Computer Applications [MCA], SJCE, Mysuru

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
II	IV	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA 441	Digital Image Processing	--

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2 Events] Marks: 50 [10* 5 Events]
2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, students will be

CO 1: Understand image formation and the role human visual system plays in perception of gray and color image data.

CO 2: Learn the signal processing algorithms and techniques in image enhancement and image restoration.

CO 3: Acquire an appreciation for the image processing issues and techniques and be able to apply these techniques to real world problems.

CO 4: To conduct independent study and analysis of image processing problems and techniques.

CO 5: Get broad exposure and understanding of various applications of image processing in industry, medicine, and defense.

TOPICS COVERED:

UNIT:1 Introduction and the fundamentals of Image Processing **12 Hours**

Origins of Digital Image Processing, examples, Fundamental Steps in Digital Image Processing,

Components of an Image Processing System, Elements of Visual Perception, A Simple Image Formation Model, Basic Concepts in Sampling and Quantization, Representing Digital Images, Zooming and Shrinking Digital Images, Some Basic Relationships Between Pixels, Linear and Nonlinear Operations.

UNIT 2: Image Enhancement using Spatial Domain **9 Hours**

Some Basic Gray Level Transformations, Histogram Processing, Enhancement Using Arithmetic/Logic Operations, Basics of Spatial Filtering, Smoothing Spatial Filters, Sharpening Spatial Filters, Combining Spatial Enhancement Methods.

UNIT 3: Image Enhancement in Frequency Domain **9 Hours**

Image Enhancement in the Frequency Domain, Introduction to the Fourier Transform and the Frequency, Domain, Smoothing Frequency-Domain Filters, Sharpening Frequency Domain Filters, Homomorphic Filtering.

UNIT 4: Image Restoration **11 Hours**

A Model of the Image degradation/Restoration process, Noise Models, Restoration in the Presence of Noise Only–Spatial Filtering, Periodic Noise Reduction by Frequency Domain Filtering, Linear, Position-Invariant Degradations , Estimating the Degradation Function, Inverse Filtering ,Minimum Mean Square Error (Wiener) Filtering.

UNIT 5: Morphology and Image Segmentation **11 Hours**

Preliminaries, Dilation and Erosion, Opening and Closing, The Hit-or-Miss Transformation, Some Basic Morphological Algorithms. Detection of Discontinuities, Edge Linking and Boundary Detection, Thresholding, Region-Based Segmentation, Patterns and Pattern Classes, Recognition Based on Decision-Theoretic Methods, Structural Methods.

TEXT BOOKS / REFERENCES

TEXT BOOK

1. Rafel C Gonzalez and Richard E. Woods: Digital Image Processing, 3rd Edition, Pearson Education, 2008.

REFERENCES:

1. K. Jain: Fundamentals of Digital Image Processing, Pearson, 2004.
2. S.Jayaraman, S.Esakkirajan, T.Veerakumar: Digital Image Processing, TataMcGraw Hill, 2011.

ADDITIONAL LEARNING SOURCES:

1. **http://www.imageprocessingplace.com/**

CO-PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO	H	H	M	M	H	L	L	L	M	L	L	L
CO	H	H	H	M	H	L	L	L	M	L	L	L
CO	H	H	H	H	M	L	M	L	M	L	L	L
CO	H	H	H	H	H	L	M	L	M	L	L	L
CO	H	H	H	H	H	M	M	M	L	L	H	M

Department of Master of Computer Applications [MCA], SJCE, Mysuru							
Course Year	**Course Semester**	**Course Type**	**Credits**	**Contact Hours/ Week**			**Total Hours/ Semester**
				Theory	**Laboratory**	**Tutorials**	
II	**IV**	**Elective**	**04**	**04**	**00**	**00**	**52**
Course No		**Course Title**				**Pre Requisites**	
MCA442		**Advanced Algorithms**				**Analysis and Design of Algorithms, Data**	

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2 Events] Marks: 50 [10* 5 Events]
2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, students will be

CO 1: Understand basic notion of algorithm, growth of functions, and recurrence equations. Implement various searching and sorting algorithms.
CO 2: Thorough understand of graph algorithms.
CO 3: Design and analyze the string matching algorithms.
CO 4: Able to implement FFT and string matching algorithms.
CO 5: Understand and design of computational geometric algorithms and its applications.

TOPICS COVERED:

UNIT:1 – Review of Analysis Techniques **12Hours**
Growth of Functions: Asymptotic notations; Standard notations and common functions; Recurrences and Solution of Recurrence equations- The substitution method, The recurrence – tree method, The master method; Amortized Analysis: Aggregate, Accounting and Potential Methods.

UNIT:2 - Graph Algorithms **10Hours**
Bellman - Ford Algorithm; Single source shortest paths in a DAG; Johnson's Algorithm for sparse graphs; Flow networks and Ford-Fulkerson method; Maximum bipartite matching.

UNIT:3 - String-Matching Algorithms: **10Hours**
Naïve string Matching; Rabin - Karp algorithm; String matching with finite automata; Knuth-Morris-Pratt algorithm; Boyer – Moore algorithms.

UNIT:4 - Polynomials and the FFT & Number -Theoretic Algorithms: **10Hours**
Representation of polynomials; The DFT and FFT; Efficient implementation of FFT. Elementary notions; GCD; Modular Arithmetic; Solving modular linear equations; The Chinese

remainder theorem; Powers of an element; RSA cryptosystem; Primality testing; Integer factorization

UNIT:5 – Computational Geometry **8Hours**

Computational geometry, approximation algorithms, parallel algorithms, Line segment properties, Vertex-cover problem, set-cover problem, PRAM model.

TEXT BOOKS / REFERENCES:

Text Books:

1. T. H Cormen, C E Leiserson, R L Rivest and C Stein: Introduction to Algorithms, 2nd Edition, Prentice-Hall of India, 2002.
2. Kenneth A. Berman, Jerome L. Paul: Algorithms, Cengage Learning, 2002.

Reference Books:

1. 1, Ellis Horowitz, Sartaj Sahni, S.Rajasekharan: Fundamentals of Computer Algorithms, University Press, 2007.

ADDITIONAL LEARNING SOURCES:

1. ocw.mit.edu/courses/electrical-engineering-and.../6...algorithms.../video-lectures
2. onlinevideolecture.com/?course_id=1311
3. iiis.tsinghua.edu.cn/~papakons/teaching/advalgorithmsF11/LectureNotes.pdf
4. https://www.cs.ucsb.edu/~suri/cs231/231.html

CO - PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
CO	H	H	M	H	H	M	H	M	M	L	M	M
CO	H	M	M	H	M	M	M	L	L	L	L	L
CO	H	M	M	M	L	L	M	L	L	L	L	L
CO	H	M	M	M	L	L	L	L	L	L	L	L
CO	H	H	M	H	L	L	M	L	M	L	L	M

Department of Master of Computer Applications [MCA], SJCE, Mysuru

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
II	IV	Theory (Elective)	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA443	Distributed Operating Systems	Operating Systems, Computer Networks.

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2 Events] Marks: 50 [10* 5 Events]
2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, students will be

CO 1: Understand a distributed system and desired properties of such systems.

CO 2: Design algorithms to handle synchronization and deadlocks in various distributed applications.

CO 3: Practice in design and implementation of selected distributed algorithms in middleware designed for group communication.

CO 4: Examine the Advanced modeling techniques for building distributed computer systems.

CO 5: Design distributed database and implementing cloud computing.

TOPICS COVERED:

UNIT:1 – Introduction to distributed systems **10 Hours**

Distributed systems-goals: advantages and disadvantages of distributed systems, hardware/ software concepts: bus-based and switched multiprocessors and multicomputers, design issues. Communications in distributed systems, layered protocols, Asynchronous Transfer Mode networks, Client- server models: client and servers, addressing, blocking, buffered, RPC, group communications: design issues, group communication in ISIS.

UNIT:2 – Synchronization in distributed systems **12 Hours**

Clock synchronization: logical clocks, physical clocks, clock synchronization algorithms and use of synchronized clocks, mutual exclusion: centralized, distributed, token ring and comparison of three algorithms, election algorithms: the bully algorithm and ring algorithm, atomic transaction: transaction model, implementation and concurrency control, deadlocks in distributed systems: distributed deadlock detection and prevention.

UNIT:3 - Processes and Processors in distributed systems **10 Hours**

Threads: usage, design issues, implementing and RPC, system models: workstation model, using idle workstation, the processor pool model and hybrid model, processor allocation: allocation models, design issues, implementation issues, examples, scheduling in distributed systems, fault tolerance: types of faults, use of redundancy, agreement in fault systems, real time distributed system: design issues, real-time communication and scheduling**.**

UNIT:4 – Distributed File Systems **10 Hours**

Design: file service interface, directory server interface, semantics of file sharing, implementation: file usage, system structure, caching, replication, example and lessons learned , trends in distributed file system: new hardware, scalability, WAN, mobile users, fault tolerance and multimedia**.**

UNIT:5 – Distributed Shared Memory **10 Hours**

Introduction, shared memory: on-chip memory, bus-based, ring-based, switched and NUMA multiprocessors and comparisons, consistency models: strict, sequential, casual, PRAM, weak,

release, entry and summary, page based distributed shared memory: basic design, replication, granularity, achieving sequential consistency, fidning the copies, finding the owner, page replacement and synchronization, shared variable distributed shared memory, object based distributed shared memory.

TEXT BOOKS / REFERENCES:

Text books:

1. Distributed Operating System, Andrew Tannenbaum, Pearson, 2013.
2. Ditributed Operating System, 2/e, Pradeep K Sinha, Pearson education, 2013.

Reference books:

1. Distributed Computing, Fundamentals, Simulations and Advanced topics, Hagit Attiya and jennifer Welch,wiley india.

ADDITIONAL LEARNING SOURCES:

1. https://www.cs.columbia.edu/~smb/classes/s06-4118/l26.pdf.
2. www.cs.helsinki.fi/u/jakangas/Teaching/DistSys/DistSys-08f-1.pdf.

CO-PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
CO	M	H	H	M	H	M	M	M	M	M	M	L
CO	H	H	H	H	H	M	M	L	M	M	L	L
CO	H	H	H	H	M	M	M	M	L	L	L	L
CO	M	H	H	H	H	M	M	M	M	M	L	L
CO	L	L	H	H	H	M	L	L	M	M	L	L

Department of Master of Computer Applications [MCA], SJCE, Mysuru							
Course Year	**Course Semester**	**Course Type**	**Credits**	**Contact Hours/ Week**			**Total Hours/ Semester**
				Theory	**Laboratory**	**Tutorials**	
II	**IV**	**Elective**	**04**	**04**	**00**	**00**	**52**
Course No		**Course Title**				**Pre Requisites**	
MCA444		**Software Project Management**				**Software Engineering concepts**	

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2 Events] Marks: 50 [10* 5 Events]
2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, students will be

CO 1: Understand the activities, evaluation and management in software project.

CO 2: Plan and different approaches for software project development.

CO 3: Can apply the software estimation and recent quality standards for evaluation of the software projects.

CO 4: Understand the risk management activities, resource allocation and monitoring and control for the projects.

CO 5: Manage people in software environments, working in teams and to maintain software quality.

TOPICS COVERED:

UNIT:1-Introduction to software project management **10 Hours**

Introduction , Why is software project management important, What is a project, Software projects versus other types of project , Contract management and technical project management ,Activities covered by software project management, Plans, methods and methodologies , Some ways of categorizing software projects, Stakeholders, Setting objectives, The business case, Project success and failure, What is management, Management control.

Project evaluation and programme management.

Introduction, A business case, Project portfolio management, Evaluation of individual projects, Cost–benefit evaluation techniques, Risk evaluation, Programme management, Managing the allocation of resources within programmes, Strategic programme, management, Creating a programme, Aids to programme management, Some reservations about, programme management, Benefits management

UNIT:2-An overview of project planning **10 Hours**

Introduction to Step Wise project, planning, Select project, Identify project scope and objectives, Identify project infrastructure, Analyse project characteristics, Identify project products and activities, Estimate effort for each activity, Identify activity risks, Allocate resources, Review/publicize plan, Execute, plan/lower levels of planning.

Selection of an appropriate project approach

Introduction, Build or buy?, Choosing methodologies and technologies, Choice of process models, Structure versus speed of delivery, The waterfall model, The spiral model, Software prototyping, Other ways of categorizing prototypes, Incremental delivery, Agile methods, Atern/Dynamic Systems Development Method, Extreme programming (XP), Managing iterative processes, Selecting the most appropriate process model .

UNIT:3-Software effort estimation **10 Hours**

Introduction, Where are estimates done?, Problems with over- and under- estimates, The basis for software estimating, Software effort estimation techniques, Bottom-up estimating, The top-down approach and parametric models, Expert judgment, Estimating by analogy, Albrecht function point analysis, Function points Mark II, COSMIC full function points, COCOMO: a parametric productivity model.

Activity planning :Introduction, The objectives of activity planning, When to plan, Project schedules, Projects and activities, Sequencing and scheduling activities, Network planning

models, Formulating a network model, Adding the time dimension, The forward pass, The backward pass, Identifying the critical path, Activity float ,Shortening the project duration, Identifying critical activities, Activity-on-arrow network.

UNIT:4-Risk management **12 Hours**

Introduction, Risk, Categories of risk, A framework for dealing with risk, Risk identification, Risk assessment, Risk planning, Risk management, Evaluating risks to the schedule, Applying the PERT technique, Monte Carlo simulation, Critical chain concepts .

Resource allocation

Introduction, The nature of resources, Identifying resource requirements, Scheduling resources ,Creating critical paths, Counting the cost, Being specific, Publishing the resource schedule ,Cost schedules, The scheduling sequence.

Monitoring and control

Introduction, Creating the framework, Collecting the data, Visualizing progress, Cost monitoring, Earned value analysis, Prioritizing monitoring, Getting the project back to target, Change control.

Managing contracts

Introduction, Types of contract, Stages in contract placement, Typical terms of a contract, Contract management, Acceptance.

UNIT:5 -Managing people in software environments **10 Hours**

Introduction, Understanding behavior, Organization behavior: a background, Selecting the right person for the job, Instruction in the best methods, Motivation, The Oldham–Hackman job characteristics model, Stress, Health and safety, Some ethical and professional concerns

Working in teams

Introduction, Becoming a team, Decision making, Organizational structures, Coordination dependencies, Dispersed and virtual teams, Communication genres, Communication plans, Leadership.

Software quality

Introduction, the place of quality in project planning, the importance of software quality, defining software quality,ISO9126,Quality management systems, process capability models, techniques to help enhance software quality, testing, quality plans.

TEXT BOOKS / REFERENCES:

Text books:

1. "Software Project Management", Bob Hughes, Mikecotterell, fifth Edition, TataMcGraw Hill, 2009.

Reference books:

1. Software Engineering Project management, Wiley Edition Second Edition edited by Richard H Thayer Foreword by Edward Yourdon
2. "Information Technology Project Management", Jack T. Marchewka, 3rd edition, Wiley India, 2009.
3. "Managing Global Projects", Ramesh, Gopalaswamy, Tata McGraw Hill, 2001.
4. "Software Project Management", Royce, Pearson Education, 1999.
5. **"**Software Project Management in Practice", Jalote, Pearson Education, 2002

ADDITIONAL LEARNING SOURCES:

1. https://mjd502.wordpress.com/study-material/notes/
2. http://www.tutorialspoint.com/software_engineering/software_project_management.htm

CO-PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
CO	M	H	H	M	H	M	M	H	M	M	H	M
CO	H	M	H	H	H	H	H	H	H	H	M	H
CO	H	H	M	M	H	M	M	H	H	M	H	M
CO	H	M	M	M	H	H	H	H	M	H	H	H
CO	M	H	H	H	H	H	H	H	M	H	M	M

Department of Master of Computer Applications [MCA], SJCE, Mysuru

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
II	IV	Elective	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA451	Cryptography and Network Security	Computer Networks-I & II

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2 Events] Marks: 50 [10* 5 Events]

2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, students will be

CO 1: Explore the need for computer security concepts.
CO 2: Understand the principles and techniques of symmetric key encryption and public key encryption.
CO 3: Describe the requirement and implementation of message authentication codes.
CO 4: Understand the concept of digital signature and key management & distribution.
CO 5: Identify the need for user authentication and its management.

TOPICS COVERED:

UNIT:1 – Introduction and Classical Encryption Technique: 9 Hours
Computer Security Concepts, OSI Security Architecture, Security Attacks, Security Services,

Security Mechanism, Model for Network Security. Symmetric Cipher Model, Substitution Techniques, Transposition Techniques.

UNIT:2 - Block Ciphers, Public Key Cryptography 10 Hours

Traditional Block Cipher Structure, The Data Encryption Standard, A DES Example, The strength of DES, Block Cipher Design Principles, AES Structure, AES Transformation Functions, AES Key Expansion, An AES Example, Principles of Public Key Cryptosystem, The RSA Algorithm, Diffie Hellman Key Exchange.

UNIT:3 - Cryptographic Hash Functions, Message Authentication Codes 12 Hours

Applications of Cryptographic Hash Functions, Two Simple Hash Functions, Requirements and Security, Hash Functions Based on Cipher Block Chaining, Secure Hash Algorithm (SHA), SHA-3, Message Authentication Requirements, Message Authentication Functions, Requirements for Message Authentication Codes, Security of MACs.

UNIT:4 –Digital Signatures, Key Management and Distribution 12 Hours

Digital Signatures, ElGamal Digital Signature, Schnorr Digital Signature Scheme, NIST Digital Signature Algorithm, Elliptic Curve Digital Signature Algorithm, RSS-PSS Digital Signature Algorithm, Comparison, Process, Services, Attacks on Digital Signature, Elgamal Digital Signature Schemes, Symmetric Key Distribution Using Symmetric Encryption and Asymmetric Encryption, Distribution of Public Keys, X.509 Certificates, Public Key Infrastructure.

UNIT:5 - User Authentication Protocols and Entity Authentication 10 Hours

Remote User Authentication Principles, Remote User Authentication Using Symmetric Encryption, Kerberos, Remote User Authentication Using Symmetric Encryption, Federated Identity Management, Personal Identity Verification, Entity Authentication: Introduction, Passwords, Challenge-Response, Zero-Knowledge, Biometrics.

TEXT BOOKS / REFERENCES:

Text books:

1. William Stallings, "Cryptography and Network Security – Principles and Practices", 6th Edition, Pearson Education, 2014.
2. Behrouz A. Forouzan and Debdeep Mukhopadhyay: "Cryptography and Network Security", 2nd Edition, Tata McGraw-Hill, 2014.

Reference books:

1. Atul Kahate, "Cryptography and Network Security" 2nd Edition, Tata McGraw-Hill Publishing Company, 2010.
2. Network Security Private Communication in a public world, Charlie Kaufman, Radia Perlman & Mike Speciner, Prentice Hall of India Private Ltd., New Delhi, 2011.
3. Network Security Essentials Applications and Standards, William Stallings, Pearson Education, New Delhi, 2010.
4. Network Security Complete Reference by Roberta Bragg, Mark Phodes-Ousley, Keith Strassberg Tata Mcgraw-Hill, 2009.

ADDITIONAL LEARNING SOURCES:

1. https://mrajacse.wordpress.com/2012/01/06/cryptography-network-security-ebooks/
2. www.williamstallings.com/Crypto/Crypto4e.html

CO-PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
CO	M	H	H	H	H	M	M	M	H	H	H	H
CO	H	H	H	H	H	H	H	L	H	M	M	M
CO	H	H	H	H	H	H	H	L	H	M	M	H
CO	H	H	H	H	H	H	M	M	H	M	H	H
CO	H	H	H	M	H	H	H	M	H	M	H	H

Department of Master of Computer Applications [MCA], SJCE, Mysuru

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
II	IV	Elective	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA452	Principles of User Interface Design	Nil

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2 Events] Marks: 50 [10* 5 Events]
2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, students will be

CO:1 Understanding the basics and features of UI Design and Development process.

CO:2 Able to evaluating interface designs, usage of software tools, Direct & Virtual manipulation of UI Components

CO:3 Able to design different Criteria for selection, Form filling & dialog boxes, Command & Natural languages.

CO:4 Able to use Interaction devices and understand the issues in UI design and able to solve them.

CO:5 Able to design effective Designing User Manuals, Online Help, Tutorials. Able to provide effective Information Search with different modes of Visualization.

TOPICS COVERED:

UNIT:1 –Introduction **12 Hours**

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 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.

UNIT:2 – Evaluating Interface Designs, Software tools, Direct & Virtual manipulation
10 Hours

Evaluating Interface Designs: Introduction, Expert Reviews, Usability Testing and Laboratories, Survey Instruments, Acceptance Tests, Evaluation During Active Use, Controlled Psychologically Oriented Experiments. Software Tools: Introduction, Specification Methods, Interface-Building Tools, Evaluation and Critiquing Tools. Direct Manipulation and Virtual Environments: Introduction, Examples of Direct Manipulation, 3D Interfaces, Tele operation, Virtual and Augmented Reality.

UNIT:3 –Menu selection, Form filling & dialog boxes, Command & Natural languages
10 Hours

Menu Selection, Form Filling, 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 Filling, Dialog Boxes, and Alternatives, Audio Menus and Menus for small Displays. Command and Natural Languages: Introduction, Functionality to Support User's Tasks, Command-Organization Strategies, the Benefits of Structure, Naming and Abbreviations, Natural Language in Computing.

UNIT:4–Interaction devices and Design Issues **10 Hours**

Interaction Devices: Introduction, Keyboards and Keypads, Pointing Devices, Speech and 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, No anthropomorphic Design, Display Design, Window Design, and Color.

UNIT:5 – User Manuals, Online Help, Tutorials, Information Search and Visualization
10 Hours

User Manuals, Online Help, 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 BOOKS / REFERENCES:

Text book:

1. Ben Shneiderman: Designing the User Interface, 4rd Edition, Pearson Education, 2009.

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
4. Introduction to GUI Design, Principles and Techniques, Wiley-Dreamtech India Pvt. Ltd, 1998.

ADDITIONAL LEARNING SOURCES:

1. http://bokardo.com/principles-of-user-interface-design/
2. http://www.ambysoft.com/essays/userInterfaceDesign.html
3. https://web.cs.wpi.edu/~matt/courses/cs563/talks/smartin/int_design.html
4. http://blog.invisionapp.com/core-principles-of-ui-design/
5. http://www.usability.gov/what-and-why/user-interface-design.html
6. https://www.ted.com/topics/interface+design

CO-PO Mapping:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
CO	H	H	H	M	M	M	M	L	M	M	M	L
CO	H	H	H	M	H	M	L	L	M	M	M	L
CO	H	H	H	H	L	M	M	L	H	L	M	L
CO	M	M	M	M	H	L	M	L	M	L	M	L
CO	H	H	H	H	H	M	M	M	H	M	L	L

Department of Master of Computer Applications [MCA], SJCE, Mysuru

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
II	IV	Elective	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA453	Enterprise Resource Planning	Management Information System

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2 Events] Marks: 50 [10* 5 Events].
2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, students will be able to,

CO 1: Understand the concept of ERP and benefits.
CO 2: Understand the related technologies of ERP.
CO 3: Understand the ERP implementation concepts.
CO 4: Understand the different business models required of ERP.
CO 5: Understand the importance and future developments of ERP.

TOPICS COVERED:

UNIT:1 - INTRODUCTION TO ERP **9 Hours**

Overview, Benefits of ERP, ERP and Related Technologies, Business Process Reengineering, Data Warehousing, Data Mining, On–line Analytical Processing, Supply Chain Management.

UNIT:2 - ERP IMPLEMENTATION **12 Hours**

Implementation Life Cycle, Implementation Methodology ,Hidden Costs, Organizing Implementation ,Vendors, Consultants and Users ,Contracts , Project Management and Monitoring.

UNIT:3 - BUSINESS MODULES **10 Hours**

Business Modules in an ERP Package, Finance, Manufacturing, Human Resource, Plant Maintenance, Materials Management, Quality Management, Sales and Distribution.

UNIT:4 - ERP MARKET **10 Hours**

ERP Market Place, SAP AG, PeopleSoft ,Baan Company , JD Edwards World Solutions Company , Oracle Corporation ,QAD , System Software Associates.

UNIT:5 - ERP – PRESENT AND FUTURE **11 Hours**

Turbo Charge the ERP System, EIA, ERP and E–Commerce, ERP and Internet, Future directions in ERP.

TEXT BOOKS / REFERENCES:

Text book:

1. Alexis Leon, "ERP Demystified", Tata McGraw Hill, 1999.

Reference books:

1. Joseph A. Brady, Ellen F. Monk, Bret J. Wangner, "Concepts in Enterprise Resource Planning", Thomson Learning, 2001.
2. Vinod Kumar Garg and N.K.Venkata Krishnan, "Enterprise Resource Planning - concepts and Planning", Prentice Hall, 1998.
3. Jose Antonio Fernandez, " The SAP R /3 Hand book", Tata McGraw Hill

ADDITIONAL LEARNING SOURCES:

1. www.scribesoft.com/ebook-the_business_case_for_integrating_erp
2. www.deltek.co.uk /erp

CO-PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
CO	L	H	M	M	M	L	L	M	M	L	H	M
CO	L	H	M	M	M	L	L	L	M	L	M	L
CO	M	M	M	L	L	M	H	M	M	M	H	M
CO	M	H	M	M	M	M	H	L	H	L	H	L
CO	L	M	M	L	M	M	M	L	H	M	H	M

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
II	IV	Elective	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA454	Computer Forensics	Operating System

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2 Events] Marks: 50 [10* 5 Events]
2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, students will be

CO 1: Identify the need for Computer Forensics and types of Computer Forensic Technology.
CO 2: Describe the types of Computer Forensic Systems.

CO 3: Illustrate the process of data recovery and determine various aspects of collecting and preserving computer evidence.
CO 4: Assess the authenticity of evidences and forensic identification.
CO 5: Estimate various ways to handle files, evidence related data and network forensics scenarios.

TOPICS COVERED:

UNIT:1 – Computer Forensic Fundamentals and types of Computer Forensic Technology 10 Hours

Introduction to computer forensics, Use of computer forensics in law enforcement, Computer forensics assistance to human resources/employment proceedings, Computer forensics services, Benefits of professional forensics methodology, Steps taken by computer forensics specialists, Who can use Computer Forensic Evidence? Types of Military Computer Forensic Technology, Types of Law Enforcement: Computer Forensic Technology, Types of Business Computer Forensic Technology.

UNIT:2 – Types of Computer Forensics Systems and Computer Forensic Services 9 Hours

Internet Security Systems, Intrusion Detection Systems, Firewall Security Systems, Biometric Security Systems, Homeland Security Systems, Occurrence of cybercrime, Cyber detectives, Fighting cyber crime with risk –management techniques, Computer forensics investigative services. Forensic process improvement.

UNIT:3 – Data Recovery, Evidence Collection and Data Seizure 10 Hours

Data Recovery Defined, Data Backup and Recovery, The role of Backup in Data Recovery, The Role of Backup in Data recovery, The Data-Recovery Solution, Hiding and recovering Hidden Data, Why Collect Evidence? Collection options, Obstacles, Types of Evidence, The Rules of

Evidence, Volatile Evidence, General Procedure, Collection and Archiving, Methods of Collection, Artifacts, Collection Steps, Controlling Contamination: The Chain of Custody, Reconstructing the Attack.

UNIT:4 – Preservation of Digital Evidence **9 Hours**

Preserving the Digital Crime Scene, Computer Evidence Processing Steps, Legal Aspects of Collecting and Preserving Computer Forensic Evidence, Special Needs of Evidential Authentication, Practical Considerations, Practical Implementation**.**

UNIT:5 – Computer Forensics Analysis **12 Hours**

Electronic document discovery: a powerful new litigation tool, Time travel, Forensics identification, Analysis of technical surveillance devices, How to become a digital detective, Useable file formats, Unusable file formats, Converting files, Network forensics scenario, A technical approach, Destruction of e-mail, Damaging computer evidence, Documenting the intrusion on destruction of data, System testing**.**

TEXT BOOKS / REFERENCES:

Text books:

1. Computer Forensics computer crime scene investigation by John R VACCA , Firewall Media,2009 edition Reprint 2012.

Reference books:

1. Guide to computer forensics and investigations by Bill Nelson, Amelia Phillips, Christopher Stuart, Cengage Learning publications, 4th edition 2013.
2. Computer Forensics by David Cowen -CISSP, Mc GrawHill education, Indian edition 2013.

ADDITIONAL LEARNING SOURCES:

1. http://study.com/articles/List_of_Free_Online_Computer_Forensics_Courses_and_Classes.html
2. https://www.udemy.com/computer-forensics-fundamentals/
3. https://forensiccontrol.com/resources/beginners-guide-computer-forensics/

CO-PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
CO	L	H	H	L	H	M	H	L	L	H	H	H
CO	L	H	H	M	H	M	H	M	M	H	H	H
CO	L	H	H	H	H	H	H	M	H	H	H	H
CO	L	H	H	H	H	H	H	H	H	H	H	H
CO	L	H	H	H	H	M	H	M	H	H	H	H

Department of Master of Computer Applications [MCA], SJCE, Mysuru							
Course Year	**Course Semester**	**Course Type**	**Credits**	**Contact Hours/ Week**			**Total Hours/ Semester**
				Theory	**Laboratory**	**Tutorials**	
II	**IV**	**Theory**	**02**	**02**	**00**	**00**	**26**
Course No		**Course Title**				**Pre Requisites**	
MCA460		**Soft Skills-IV [Research Methodology & Seminar]**				**Nil**	

COURSE ASSESSMENT METHOD:

1. Internal Assessment 5 Events [3 Written Tests, Events: 2 Seminars] Marks: 50 [10* 5 Events].
2. Semester End Exam [50 Marks, 1½ Hours].

COURSE OUTCOMES:

Upon completion of this course students are able to,

CO 1: Understand the basics of research methodology and its importance in research

CO 2: Define research problem and design the research.

CO 3: Learn to design the sample for the research and understand different techniques.

CO 4: Understand and use different techniques for collection data

CO 5: Interpret, analyze and document the findings of the research or work.

TOPICS COVERED:

UNIT:1 – Introduction to Research Methodology **6 Hours**

Introduction, Meaning of research, Objectives of research, Motivation in research, Types of research, Research approaches, Significance of research, research methods v/s methodology, research and scientific methods, Importance of know how research is done, research process, criteria for good research, Problems

UNIT:2 – Defining Research problem and Research Design **6 Hours**

What is research problem, Selecting the research, Necessity of defining the problem, technique involved in defining the problem, a illustration. Research Design: Meaning of research design, Need for research design, features of good design, importance concepts relating to research design, different research designs, basic principles of experimental designs.

UNIT:3 – Sampling design **4 Hours**

Census and Sample Survey, Implications of a Sample Design, Steps in Sampling Design, Criteria of Selecting a Sampling Procedure, Characteristics of a Good Sample Design, Different Types of Sample Designs, How to Select a Random Sample?, Random Sample from an Infinite Universe, Complex Random Sampling Designs

UNIT:4 - Methods of Data Collection **4 Hours**

Collection of Primary Data, Observation Method, Interview Method, Collection of Data through Questionnaires, Collection of Data through Schedules, Difference between Questionnaires and Schedules, Some Other Methods of Data Collection, Collection of Secondary Data Selection of Appropriate Method for Data Collection, Case Study Method.

UNIT:5 - Interpretation and Report Writing **6 Hours**

Meaning of Interpretation, Why Interpretation?, Technique of Interpretation, Precaution in Interpretation, Significance of Report Writing, Different Steps in Writing Report, Layout of the Research Report, Types of Reports, Oral Presentation, Mechanics of Writing a Research Report, Precautions for Writing Research Reports.

TEXT BOOKS / REFERENCES:

Text book:

1. Ranjith kumar, Research Methodology: a step by step guide for beginners, 3rd edition, Sage publications Ltd. 2011.

Reference book:

1. C. R. Kothari and Gaurav Garg, Research Methodology: Methods & Techniques, 3rd edition, New age international publications, 2013.

ADDITIONAL LEARNING SOURCES:

1. https://explorable.com/research-methodology
2. https://www.sophia.org/tutorials/research-methods--9
3. http://edutechwiki.unige.ch/en/Methodology_tutorial

CO - PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO1	PO1	PO1
CO1	L	M	M	M	M	H	M	L	M	L	M	M
CO2	L	H	M	H	M	H	M	L	M	L	L	L
CO3	L	M	L	H	M	M	M	L	M	L	L	L
CO4	L	H	L	H	M	M	L	L	H	L	L	L
CO5	L	M	L	M	M	H	L	L	H	L	L	L

Department of Master of Computer Applications [MCA], SJCE, Mysuru

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
II	III	Theory	04	00	03	00	48

Course No	Course Title	Pre Requisites
MCA41L	Web Programming laboratory	Unix, Computer Concepts

COURSE ASSESSMENT METHOD:

1. Internal Assessment [Test-1+ Test-2 + Observation book + Record + Viva: 15 + 15 + 05 + 05 +10] Marks: 50.

COURSE OUTCOMES:

Upon successful completion of this course, students will be

CO 1: Acquire knowledge about functionalities of world wide web.

CO 2: Design of client side validation using scripting languages.

CO 3: Able to design front end web page and connect to end database

LIST OF EXPERIMENTS COVERED:

1. Develop and demonstrate a XHTML document that illustrates the use external style sheet, ordered list, table,
Borders, Padding, color, and the tag.

2. Develop and demonstrate a XHTML file that includes JavaScript script for the following problems:
a) Input: A number n obtained using prompt.
Output: The first n Fibonacci numbers.
b) Input: A number n obtained using prompt.
Output: A table of numbers from 1 to n and their squares using alert.

3. Develop and demonstrate a XHTML file that includes JavaScript script that uses functions for the
following problems:
a) Parameter: A string Output: The position in the string of the left-most vowel.
b) Parameter: A number Output: The number with its digits in the reverse order.

4. a) Develop and demonstrate, using JavaScript script, 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 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.
b) Modify the above program to get the current semester also (restricted to be a number from 1 to 8).

5. a) 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.
b) Modify the above document so that when a paragraph is moved from the top stacking position, it returns to its original position rather than to the bottom.

6. a) Design an XML document to store information about a student in an engineering College affiliated to

VTU. The information must include USN, Name, Name of the College, Brach, 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.

7. a) Write a Perl program to display various Server Information like Server Name, Server Software, Server protocol, CGI revision etc.

b) Write a Perl program to accept UNIX command from a HTML form and to display the output of the command executed.

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

b) 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.

9. Write a Perl program to display a digital clock which displays the current time of the server.

10. 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.

11. 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.

12. Write a PHP program to store page views count in SESSION, to increment the count on each refresh, and to show the count on web page.

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. Using PHP and MySQL, develop a program to accept book information viz. Accession number, title, authors, edition and publisher from a web page and store the information in a database and to search for a book with the title specified by the user and to display the search results with proper headings.

TEXT BOOKS / REFERENCES:

Text books:

1.Harvey & Paul Deitel & Associates, Harvey Deitel and abbey Deitel, "Internet and World Wide Web, Fift Edition, Pearson Education,2011.

2. Mike Mcgrath, "PHP & MySQL in easy steps", Tata McGraw Hill,2012.

ADDITIONAL LEARNING SOURCES:

1. http://php.net/manual/
2. www.webbasedprogramming.com

CO-PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
CO	M	M	M	M	M	L	M	L	L	L	M	M
CO	M	M	M	M	M	L	M	L	L	L	M	M
CO	M	M	M	M	M	L	M	L	L	L	M	M

Department of Master of Computer Applications [MCA], SJCE, Mysuru

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
II	IV	Laboratory	01	00	03	00	48

Course No	Course Title	Pre Requisites
MCA42L	Advanced Java Programming Laboratory	Java Programming

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 2 Written Tests, Viva-voce + Record + Observation] Marks: 50 [15+15+10+5+5]

COURSE OUTCOMES:

Upon successful completion of this course, students will be able to,

CO 1: Understand Enterprise Application Architecture and J2EE, and develop Database applications using JDBC.
CO 2: Understand the concept of Server side applications using Servlets and develop server side applications.
CO 3: Understand JSP and write applications using JSP
CO 4: Design and develop components using Java Beans and understand Annotations
CO 5: Apply the concept of middle & Data Based application layers for an enterprise application using Enterprise Java Beans (EJB) and learn to develop the Server Side Component Models.

LIST OF PROGRAMS TO BE COVERED:

1. Create a Static web page using HTML.
2. Write a JDBC program to establish connection to a Database
3. Write a JDBC program to INSERT records to the Database using Statement Object
4. Write a JDBC program to UPDATE the record of a Database using Statement Object
5. Write a JDBC program to DELETE the record from a Database using Statement Object
6. Write a JDBC program to INSERT records to the Database using PrepareStatement Object
7. Write a JDBC program to UPDATE the record of a Database using PrepareStatement Object
8. Write a JDBC program to DELETE the record from a Database using PrepareStatement Object
9. Write a JDBC program to INSERT records to the Database using Statement Object With Batch Insert

10. Write a JDBC program to UPDATE the record of a Database using Statement Object with Batch Update
11. Write a JDBC program to DELETE the record from a Database using Statement Object with Batch Delete
12. Write a JDBC program to INSERT records to the Database using PrepareStatement Object with Batch Insert
13. Write a JDBC program to UPDATE the record of a Database using PrepareStatement Object with Batch Update
14. Write a JDBC program to DELETE the record from a Database using PrepareStatement Object with Batch Delete
15. Write a servlet program to display hello world.
16. Write a servlet program to generate a plain text and HTML output.
17. Write a servlet program to generate lottery numbers
18. Write a servlet program to demonstrate single thread model.
19. Write a servlet program to read three parameters from form Data
20. Write a servlet program to read all data from form data.
21. Write a servlet program to demonstrate response header
22. Write a servlet program to show request header.
23. Write a servlet program to display number of visitors visited the page.
24. Write a servlet program to handle cookies.
25. Write a Servlet program to demonstrate session tracking
26. Create a simple Login application using HTML, MySQL, Servlet and Web-XML.
27. Create a simple User Registration application using HTML, MySQL, Servlet and Web-XML.
28. Write a Servlet program to generate a output in different MIMEs (Excel, HTML, PDF etc.)
29. Write a JSP program to read three parameters from another JSP page and display.
30. Write a JSP program to display Sales entry information.
31. Write a JSP program to test the background color of the page
32. Write a JSP program to generate access counter.
33. Write a JSP program which conditionally generates a spreadsheet
34. Write a program to generate a Plug in Applet using JSP
35. Write a program to generate a string bean using JSP
36. Write a program to create a Excel sheet using JSP
37. Write a program to create a Bean sheet using
38. Write a program to create a reusable footer using JSP
39. Write a program to count the shared counts using JSP.
40. Write a Program which includes files using JSP
41. Write a JSP program which generates the wishes for the day based on time.
42. Write a JSP program to generate random numbers.
43. Write a JSP program to generate a order confirmation by taking the order details from the user.
44. Write a JSP program which should compute the speed.
45. Create a Java Bean component for generating the access counter for the page visit on Click Button event. Use this bean component as reference to a new project and demonstrate the functionality of the Bean and its reusability.
46. Create a simple application using JSP, Servlets, JDBC, MIME, Java Bean and MySQL to accept the details from the student and display the marks list for the student with grade (Reference – your previous semester marks card)
47. Design and develop any application of your own using EJB Componets (Session, Messhae

Driven and Entity Bean)

TEXT BOOKS / REFERENCES:

Text Books:

1. Marty Hall, Larry Brown. Core Servlets and Java Server Pages. Volume 1: Core Technologies. Second Edition. (Chapter 3,4,5,6,7,8,9,10,11,12,13,14).
2. Java 6 Programming Black Book, Dreamtech Press. 2012 (Chapter 17,18,19,20,21,22,27,28,29,30).
3. Andrew 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, Eight Edition. Comprehensive coverage of the Java Language. Tata McGraw-Hill Edition – 2011.

ADDITIONAL LEARNING SOURCES:

1. www.j2eetutorials.50webs.com
2. www.docs.oracle.com/javaee/6/tutorial/doc/
3. www.tutorialspoint.com/listtutorials/java/j2ee/1
4. www.coderanch.com/t/534906/EJB-JEE/java/Tutorial-EE-Beginners
5. www.j2eebrain.com/
6. http://w3schools.invisionzone.com/index.php?showforum=52
7. http://www.javagenious.com/search/jsp-tutorial-w3schools/
8. http://gsraj.tripod.com/java/java_tutorial.html

CO - PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
CO	H	H	H	H	H	H	H	M	M	M	H	M
CO	H	M	H	M	H	M	H	L	M	M	M	L
CO	H	M	H	M	H	M	H	L	M	M	M	L
CO	H	M	M	M	H	M	H	L	M	M	M	L
CO	H	H	H	H	H	H	H	M	H	M	H	H

Course Year	**Course Semester**	**Course Type**	**Credits**	**Contact Hours/ Week**			**Total Hours/ Semester**
				Theory	**Laboratory**	**Tutorials**	
II	**IV**	**Seminar**	**02**	**00**	**00**	**00**	**32**

Course No	**Cours Title**	**Pre Requisites**
MCA46S	**Seminar**	**NIL**

COURSE ASSESSMENT METHOD:

- Internal Assessment (4 events - presentation and report) = 50 Marks.

COURSE OUTCOMES:

Upon successful completion of the seminar, the student will be able to;

1. Explore the latest trends related to computer science and technology.
2. Acquire knowledge beyond curriculum.
3. Examine and record the prospects of the topic of interest.
4. Understand and contemplate the topic efficiently.
5. Document and present the topic effectively.

Guidelines:

1. Student has to choose advanced /current topics from publication such as IEEE explore/IEEE journals Science Direct/ElseVier.com/ journals from reputed publishers/reputed books /Conference publications.
2. Students are allocated with individual guides.
3. Students have to finalize the seminar topics in consultation with guides.
4. After getting approval from the guide students has to submit the synopsis to their respective guide.
5. Submitted synopsis needs to be presented before the panel for approval .
6. Students need to incorporate suggestions given by the panel before the final Presentation.
7. Students need to present the progress of the work in mid of the semester.
8. At the end of the semester, the panel will check the significance of the topic to the current Scenario, the presentation skill, the question and answers and report generated.
9. For final presentation students needs to present the full topic with a report before the panel.
10. Students need to present the topic with all relevant data and information.
11. Presentation will be evaluated base on the following parameters:
    i. Relevance of the topic to the current technology.
    ii. Presentation skill.
    iii. Question and Answers.
    iv. Report written
12. Students need to follow the guidelines given below while writing the report.
    i. Report Prefer Font style-Times New Roman,
    ii. Heading Size: 16 Bold,
    iii. Sub Heading Size: 14 Bold,
    iv. Body: 12 Normal,
    v. Page Layout: Normal,
    vi. Line Space: 1.15,
    vii. Alignment: Both Sides.
13. Format for submission of the synopsis:
    i. Title of the seminar.
    ii. Objective.
    iii. Introduction.
    iv. Brief description.
14. Format for submission of the final seminar report:
    i. Title of the seminar.
    ii. Objective.
    iii. Introduction.
    iv. Detail description of the topic with all relevant data and information.

v. Conclusion.

15. Evaluation procedure:

• Topic selection/Presentation	10 Marks
• Midterm evaluation	10 Marks.
• Attendance of the student	05 Marks
• Report submitting in the given format. Final presentation (presentation skills) Question and Answers.	10 Marks 10 Marks 05 Marks.
Total (A + B + C + D)	50 Marks

V Semester

Department of Master of Computer Applications [MCA], SJCE, Mysuru.							
Course Year	**Course Semester**	**Course Type**	**Credits**	**Contact Hours/ Week**			**Total Hours/ Semester**
				Theory	**Laboratory**	**Tutorials**	
III	**V**	**Theory**	**04**	**04**	**00**	**00**	**52**
Course No		**Course Title**				**Pre Requisites**	
MCA510		**Python Programming**				**OOPS**	

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2 Events] Marks: 50 [10* 5 Events]
2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, students will be able to:

CO1: Understand and comprehend the basics of python programming.

CO2: Apply knowledge in real time applications.

CO3: Understands about files and its applications.

CO4: Apply object-oriented programming concepts to develop dynamic interactive Python applications

CO5: Demonstrate MVC architecture with respect to Django framework

TOPICS COVERED:

UNIT 1 - Introduction to Python **10 Hours**

Python Basics: Data Types, Operators, Input/Output Statements, Creating Python Programs, Python Flow Control statements: Decision making statements, Indentation, Conditionals, loops, break, continue, and pass statements. Strings, lists, Tuples, Dictionaries

UNIT 2 - Python Functions **10 Hours**

Defining functions, DOC strings, Function parameters: default, keyword required and variable length arguments, key-word only parameters, local and global variables, pass by reference versus value, Anonymous functions, Recursion. Functional Programming:
Mapping, Filtering and Reduction, Lambda Functions, List Comprehensions.

UNIT 3 - Object Oriented Programming **10 Hours**

Definition and defining a class, Constructor, Destructor, self and del keywords, Access to Attributes and Methods, geattr and hasattr attributes, Data Attributes and Class Attributes, Data Hiding, Inheritance, Static Members. Regular Expressions: Defining Regular Expressions and String Processing.

UNIT 4 - File Handling and Python GUI Programming **10 Hours**

File object attributes, Read and Write into the file, Rename and Delete a File, Exceptions Handling: Built-in Exceptions and User defined Exceptions GUI Programming, Introduction to Python GUI Programming, Tkinter Programming, Tkinter widgets, Events and Bindings

UNIT 5 - Working with Django **12 Hours**

Rendering Templates into HTML and Other Formats, Understanding Models, Views, and Templates, Separating the Layers (MVC) - Models, Views, Templates, Overall Django Architecture, Defining and Using Models, Using Models, Templates and Form Processing, Setting up the Database, Using a Database Server, Using SQLite, Creating the Tables.

TEXT BOOKS / REFERENCES:

Text books:

1. Timothy A. Budd: Exploring Python, Tata McGraw-Hill, 2011.
2. Jeff Forcier, Paul Bissex, Wesley Chun: Python Web Development with Django, Addison-Wesley, 2008.

Reference books:

1. Ascher, Lutz: Learning Python, 4th Edition, O'Reilly, 2009.

2. Wesley J Chun: Core Python Applications Programming, Pearson Education, 3rd Edition, 2013.

3. Paul Gries, Jennifer Campbell, Jason Montojo , Practical Programming: An introduction to Computer Science Using Python, second edition, Pragmatic Bookshelf.

4. Allen Downey , Jeffrey Elkner , Learning with Python: How to Think Like a Computer Scientist Paperback –, 2015 **.**

ADDITIONAL LEARNING SOURCES:

http://www.network-theory.co.uk/docs/pytut/
http://docs.python.org/tutorial/
http://zetcode.com/tutorials/pythontutorial/
http://www.sthurlow.com/python/ http://www.tutorialspoint.com/python/
http://www.djangoproject.com/
http://www.djangobook.com/

CO-PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO1
CO	H	L	L	M	L	L	L	L	M	L	L	L
CO	L	M	M	M	H	L	L	L	M	H	L	L
CO	H	H	H	H	M	M	L	L	L	L	L	L
CO	H	H	H	H	M	M	L	L	L	L	L	L
CO	L	H	L	L	H	M	L	L	L	L	L	L

Department of Computer Applications [MCA], SJCE, Mysuru.							
Course Year	**Course Semester**	**Course Type**	**Credits**	**Contact Hours/ Week**			**Total Hours/ Semester**
				Theory	**Laboratory**	**Tutorials**	
III	**V**	**Theory**	**05**	**04**	**00**	**02**	**52**

Course No	**Course Title**	**Pre Requisites**
MCA520	**Data Mining and Analytics**	**Database management systems**

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, Any 2 Events from the list [Test, Presentation, Quiz. Projects, Group discussion, assignments, Viva-Voce etc.]

Marks: 50 [10* 5 Events].

2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, the student will be able to:
CO1: Interpret the basic concepts, principles and techniques of data mining.
CO2: Define knowledge discovery and data mining; recognize the key areas and issues in data mining.
CO3: To apply the techniques of clustering, classification, association finding, feature selection and visualization of real world data.
CO4: Understand and implement the basic concept and principles of data analytics.
CO5: Apply evaluation metrics to select appropriate techniques in data analytics.

TOPICS COVERED:

UNIT:1 - Introduction **12 Hours**

Introduction to Data Warehouse. A Multidimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, Data Cube Technology, From Data warehousing to Data Mining, Data Mining, Functionalities, Data Cleaning, Data Integration and Transformation, Data Reduction. Data Mining Primitives, languages, and system Architectures, A Data Mining Query Language.

UNIT:2 - Mining Association Rules in Large Data Bases **10 Hours**

Association Rule Mining Single-Dimensional Boolean Association Rules from Transactional Databases, Mining Multilevel Association Rules from Transactional Databases.

UNIT:3 - Classification and Prediction **10 Hours**

Issues regarding Classification and Prediction, Classification by Decision tree induction, Bayesian Classification, Classification by Back propagation, Classification based on the concepts from association rule mining, Other classification methods, Prediction.

UNIT:4 - Data Analysis **10 Hours**

Evolution Of Analytic Scalability – Convergence – Parallel Processing Systems – Cloud Computing – Grid Computing – Map Reduce – Enterprise Analytic Sand Box – Analytic Data Sets – Analytic Methods –

Analytic Tools – Cognos – Microstrategy - Pentaho. Analysis Approaches – Statistical Significance – Business Approaches – Analytic Innovation – Traditional Approaches – Iterative

UNIT:5 - Cluster Analysis **10 Hours**

What is Cluster Analysis? Types of data in Cluster Analysis: A Categorization of Major Clustering Methods. Partitioning Methods, Hierarchical Methods, Density-Based Methods, Model Based Clustering Methods: Statistical Approach, Neural Network Approach. Outliner Analysis.

TEXT BOOKS / REFERENCES:

Text books :

1. Jiawei, Micheline Kamber, "Data Mining Concepts and Techniques", Morgan Kauf Mann Publishers.2012

2. Bill Franks, Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with advanced analytics, John Wiley & sons, 2012.

Reference Books :

1. Arun.K.Poojari, "Warehousing and Mining", PHI 2010.

2. Michael, J.Berry, Gordon Linoff: "Data Mining Techniques: Marketing, Sales, Customer support", John wiley & sons, 2010.

3. Paul Zikopoulos, Chris Eaton, Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data: Analytics for Enterprise Class Hadoop and Streaming Data, McGraw Hill Professional, 2011.

4. Michael Berthold, David J. Hand, Intelligent Data Analysis, Springer, 2007**.**

ADDITIONAL LEARNING SOURCES:

1. web.cse.ohio-state.edu/~srini/674/part1.ppt.
2. www.cse.iitb.ac.in/~dbms/Data/Talks/datamining-intro-IEP**.**

CO-PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	H	M	M	H	M	L	M	M	M
CO2	H	M	M	H	M	M	H	L	L	M	L	L
CO3	M	M	M	H	M	H	H	M	L	M	L	L
CO4	M	M	M	H	M	M	L	L	M	M	L	L
CO5	H	H	M	H	M	M	M	M	L	H	L	L

Department of Master of Computer Applications [MCA], SJCE, Mysuru							
Course Year	**Course Semester**	**Course Type**	**Credits**	**Contact Hours/ Week**			**Total Hours/ Semester**
				Theory	**Laboratory**	**Tutorials**	
III	**V**	**Theory**	**04**	**04**	**00**	**00**	**52**

Course No	**Course Title**	**Pre Requisites**
MCA530	**Software Architecture**	**Software Engineering**

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2 Events] Total Marks: 50 [10* 5].
2. Semester End Exam [Total Marks:100 Marks, Duration: 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, the student will be able to:

CO1 : Understand the need and importance of software architectures.
CO2 : Achieve system qualities like performance, security, and maintainability.
CO3 : Learn the system requirements and Choose a single or set of appropriate architectures based on the requirement.
CO4 : Evaluate the selected architecture for feasibility, Manage complexity, and Achieve reusability.
CO5 : Understand and achieve different tactics and Document the architectures.

TOPICS COVERED:

UNIT 1 - Introduction, Architectural Styles **12 Hours**

The Architecture Business Cycle: Where do architectures come from? Software processes and the architecture business cycle; What makes a "good" architecture? What software architecture is and what it is not; Other points of view; Architectural patterns, reference models and reference architectures; Importance of software architecture; Architectural structures and views.

Architectural styles; Pipes and filters; Data abstraction and object-oriented organization; Event-based, implicit invocation; Layered systems; Repositories; Interpreters; Process control; Other familiar architectures; Heterogeneous architectures.

UNIT 2 - Understanding and Achieving Quality Attributes **10 Hours**

Functionality and architecture; Architecture and quality attributes; System quality attributes; Quality attribute scenarios in practice; Other system quality attributes; Business qualities; Architecture qualities.

Achieving Quality: Introducing tactics; Availability tactics; Modifiability tactics; Performance tactics; Security tactics; Testability tactics; Usability tactics; Relationship of tactics to architectural patterns; Architectural patterns and styles**.**

UNIT 3 - Architectural Patterns – From Mud to Structures, Distributed Systems **12 Hours**

Introduction: From mud to structure: Layers, Pipes and Filters, Blackboard. Distributed Systems: Broker; Interactive Systems: MVC, Presentation-Abstraction-Control.

UNIT 4 - Adaptable Systems & Other systems **08 Hours**

Adaptable Systems: Microkernel; Reflection. Structural decomposition: Whole – Part; Organization of work: Master – Slave; Access Control: Proxy.

UNIT 5 - Designing and Documenting Software Architecture **10 Hours**

Architecture in the life cycle; Designing the architecture; Forming the team structure; Creating a skeletal system. Uses of architectural documentation; Views; Choosing the relevant views; Documenting a view; Documentation across views.

TEXT BOOKS / REFERENCES:

Text books:

1. Len Bass, Paul Clements, Rick Kazman: Software Architecture in Practice, 3d Edition, Pearson Education, 2013.
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, 2012.
3. Mary Shaw and David Garlan: Software Architecture -Perspectives on an Emerging Discipline, Prentice Hall of India, 2010.

Reference books:

1. Richard N. Taylor, Nenad Medvidovic and Eric M. Dashofy: Software Architecture: Foundations, Theory, and Practice, Wiley- India 2012.

ADDITIONAL LEARNING SOURCES:

1. http://www.sei.cmu.edu/architecture/
2. http://handbookofsoftwarearchitecture.com/
3. https://leanpub.com/software-architecture-for-developers/read
4. http://www.hillside.net/patterns/

CO-PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
CO	H	H	H	H	H	H	H	M	M	M	H	M
CO	H	M	H	M	M	M	H	L	M	M	M	M
CO	H	M	H	M	M	M	H	H	M	M	M	M
CO	H	M	M	M	M	M	H	L	M	M	M	H
CO	M	H	H	H	H	H	H	M	H	M	H	H

JSS Science and Technology University, Mysuru
Department of Master of Computer Applications [MCA]

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
III	V	Theory (Elective)	04	04	00	00	52
Course No		Course Title				Pre Requisites	

MCA545	Big Data	Data Warehousing and Mining

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2 Events]Marks: 50 [10* 5 Events]
2. Semester End Exam [100 Marks, 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, students will be

CO1: Understand the significance, structure and sources of Big data.
CO2: Asses avenues for analytical scalability.
CO3: Comprehend stream computing and applications.
CO4: Apply the different clustering techniques.
CO5: Use different Frame works and Visualization techniques.

TOPICS COVERED:

Unit 1 - Introduction to Big Data **12Hours**

What is big data? Is the "big" part or the "data" art more important? How is big data different? How is big data more of the same? Risks of big data -why you need to tame big data -the structure of big data- exploring big data, most big data doesn't matter- filtering big data effectively -mixing big data with traditional data- the need for standards-today's big data is not tomorrow's big data. Web data: the original big data -web data overview -what web data reveals -web data in action? A cross-section of big data sources and the value they hold.

Unit 2 - Data Analysis **08 Hours**

Evolution of analytic scalability,convergence ,parallel processing systems ,cloud computing, grid computing,map reduce , enterprise analytic sand box ,analytic data sets ,analytic methods , analytic tools,Analysis approaches,statistical significance, business approaches,analytic innovation.

Unit 3 - Mining Data Streams **12 Hours**

The stream data model -A Data-Stream-Management System,Examples of Stream Sources, Stream Queries, Issues in Stream Processingsampling data in a stream- A Motivating Example, a Representative Sample, The General Sampling Problem, Varying the Sample Size , filtering streams-A Motivating Example, Bloom Filter, Analysis of Bloom Filtering, counting distinct elements in a stream, The Count-Distinct Problem,The Flajolet-Martin Algorithm, Combining Estimates, Requirements ,estimating moments, Definition of Moments, Alon-Matias-Szegedy Algorithm for Second Moments,the Alon-Matias-Szegedy Algorithm Works, Higher-Order Moments, Dealing With Infinite Streams ,counting ones in a window-The Cost of Exact Counts, TheDatar-Gionis-Indyk-Motwani Algorithm, Storage Requirements for the DGIM Algorithm, Query Answering in the DGIM Algorithm, Maintaining the DGIM Condition, Reducing the Error, Extensions to the Counting of Ones, decaying window-The Problem of Most-Common Elements, Definition of the Decaying Window, Finding the Most Popular Elements .

Unit 4 - Frequent Item sets and Clustering **12 Hours**

Mining frequent itemsets, market based model ,apriorialgorithm,handling large data sets in main memory,limited pass algorithm, counting frequent itemsets in a stream,clustering techniques, hierarchical clustering – k- means algorithm,the CURE algorithm,clustering in non-euclidean space,clustering for streams and parallelism**.**

Unit 5 :Mining Social-Network Graphs **08 Hours**

Social Networks as Graphs,What is a Social Network,Social Networks as Graphs, Varieties of Social Networks, Graphs With Several Node Types,Clustering of Social-Network Graphs, Distance Measures for Social-Network Graphs,applying Standards for Clustering Methods, Betweennes, The Girvan-Newman Algorithm, Using Betweenness to Find Communities, Direct Discovery of Communities,FindingCliques,Complete Bipartite Graphs, Finding Complete Bipartite Subgraphs,Why Complete Bipartite Graphs Must Exist .

TEXT BOOKS / REFERENCES:

Text books:

1. Bill Franks, Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with advanced analytics, John Wiley & sons, 2013.
2. Rajaraman and Jeffrey David Ullman, Mining of Massive Datasets, Cambridge University Press, 2014

Reference books:

1. Paul Zikopoulos, Chris Eaton, Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data: Analytics for Enterprise Class Hadoop and Streaming Data, McGraw Hill Professional, 2012.
2. Glenn J. Myatt, Making Sense of Data, John Wiley & Sons, Pete Warden, Big Data Glossary, O‟Reilly.
3. Chuck Lam, “Hadoop in Action”, Dreamtech Press.

ADDITIONAL LEARNING SOURCES:

1. https://www.tutorialspoint.com/big_data_tutorials.html.
2. https://www.lynda.com/Big-Data-training-tutorials/2061-0.html.
3. https://www.tutorialspoint.com/hadoop/hadoop_big_data_overview.html.
4. https://bigdatauniversity.com.

CO-PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
CO	M	H	H	H	M	M	M	L	L	M	H	H
CO	H	H	H	M	M	M	M	L	L	M	H	M
CO	H	H	M	H	M	M	M	L	L	M	H	M
CO	H	H	M	H	H	M	L	L	M	L	H	M
CO	H	H	H	M	H	M	L	L	M	L	M	M

Department of Master of Computer Applications [MCA], SJCE, Mysuru.							
Course Year	**Course Semester**	**Course Type**	**Credits**	**Contact Hours/ Week**			**Total Hours/ Semester**
				Theory	**Laboratory**	**Tutorials**	
III	**V**	**Theory**	**04**	**04**	**00**	**00**	**52**
Course No		**Course Title**				**Pre Requisites**	

MCA554	.Net Programming	OOPS, HTML Concepts

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2 Events] Total Marks: 50 [10* 5].
2. Semester End Exam [Total Marks:100 Marks, Duration: 3 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, the student will be able to:

CO1 : Understand the concept of .NET framework and basics of C# .NET.
CO2 : Create Console Applications using the C#.NET.
CO3 : Develop Web Applications using the ASP.NET
CO4 : Understand the ASP.NET web form, state management and error handling mechanism
CO5 : Access and manipulate data in a database by using Microsoft ADO.NET

TOPICS COVERED:

Unit – 1 **10 Hours**

Getting started with .NET Framework 4.0:- Benefits of .NET Framework, Architecture of .NET Framework 4.0, Components of .NET Framework 4.0: CLR, CTS, Metadata and Assemblies, .NET Framework Class Library, Windows Forms, ASP .NET and ASP .NET AJAX, ADO .NET, Windows workflow Foundation, Windows Presentation Foundation, Windows Communication Foundation, Widows Card Space and LINQ.

Introducing C#:- Need of C#, C# Pre-processor Directives, Creating a Simple C# Console Application, Identifiers and Keywords. Data Types, Variables and Constants: Value Types, Reference Types, Type Conversions, Boxing and Unboxing, Variables and Constants .Expression and Operators : Operator Precedence, Using the ?? (Null Coalescing) Operator, Using the:: (Scope Resolution) Operator and Using the is and as Operators. Control Flow statements: Selection Statements, Iteration Statements and Jump Statements**.**

Unit – 2 **10 Hours**

Namespaces, Classes, Objects and Structures:- Namespaces, The System namespace, Classes and Objects: Creating a Class, Creating an Object, Using this Keyword, Creating an Array of Objects, Using the Nested Classes, Defining Partial Classes and Method, Returning a Value from a Method and Describing Access Modifiers. Static Classes and Static Class Members. Properties: Read-only Property, Static Property, Accessibility of accessors and Anonymous types. Indexers, Structs: Syntax of a struct and Access Modifiers for structs.

Object- Oriented Programming:- Encapsulation: Encapsulation using accessors and mutators, Encapsulation using Properties. Inheritance: Inheritance and Constructors, Sealed Classes and Sealed Methods, Extension methods. Polymorphism: Compile time Polymorphism/ Overloading, Runtime Polymorphism/ Overriding. Abstraction: Abstract classes, Abstract methods. Interfaces: Syntax of Interfaces, Implementation of Interfaces and Inheritance

Unit – 3 **14 Hours**

Delegates and Events and Exception Handling:- Delegates: Creating and using Delegates, Muticasting with Delegates. Events: Event Sources, Event Handlers, Events and Delegates, Multiple Event Handlers. Exception Handling: The try/catch/finally statement, Checked and Unchecked Statements.

Graphical User Interface with Windows Forms:- Introduction, Windows Forms, Event Handling: A Simple Event- Driven GUI, Visual Studio Generated GUI Code, Delegates and Event- Handling Mechanism, Another Way to Create Event Handlers, Locating Event Information. Control Properties and Layout, Labels, TextBoxes and Buttons, GroupBoxes and Panels, CheckBoxes and RadioButtons, ToolTips, Mouse-Event Handling, Keyboard-Event Handling. Menus, MonthCalendar Control, Date TimePicker Control, LinkLabel Control, ListBox Control, CheckedListBox Control, ComboBox Control, TreeView Control, ListView Control, TabControl Control and Multiple Document Interface (MDI) Windows

Unit – 4 **09 Hours**

Data Access with ADO.NET:- Understanding ADO.NET: Describing the Architecture of ADO.NET, ADO>NET,ADO.NET Entity Framework. Creating Connection Strings: Syntax for Connection Strings. Creating a Connection to a Database: SQL Server Database, OLEDB Database, ODBC Data Source. Creating a Command Object. Working with DataAdapters: Creating DataSet from DataAdapter, Paging with DatyaAdapters, Updating with DataAdapters, Adding Multiple Tables to a DataSet, Creating Data View. Using DataReader to Work with Databases**.**

Unit – 5 **09 Hours**

Web App Development with ASP.NET:- Introduction, Web Basics, Multitier Application Architecture, Your First Web Application: Building WebTime Application, Examining WEebTime.aspx's Code-Behind File, Standard Web Controls: Designing a Form, Validation Controls, Session Tracking: Cookies, Session Tracking with http Session State, Optaions.aspx: Selecting a Programming Language, ecommenations.aspx: Displaying Recommendations based on Session Values. Case study: Database-Driven ASP.NET Guestbook, Building a Web Form that Displays Data from a Database, Modifying the Code-Behind File for the Guestbook Application, ASP.NET AJAX: Traditional Web Applications, Ajax Web Applications, Testing an ASP.NET Ajax application, the ASP.NET Ajax Control Toolkit. Case study: Password-Protected Books Database Application.

TEXT BOOKS / REFERENCES:

Text Books:

1. .NET 4.0 Programming (6-in-1), Black Book, Kogent Learning Solutions Inc., Wiely- Dream Tech Press. (Chapters: 1,10,11,12,13,14 and 19).

2. Paul Deitel and Harvey Deitel: C# 2010 for Programmers, 4th Edition, Pearson Education. (Chapters: 14,15,19 an 27.3)

References Books:

1. Andrew Trolsen: Pro C# 5.0 and the .NET 4.5 Framework, 6th Edition, Wiely-Appress.

2. Bart De Smet: C# 4.0 Unleashed, Pearson Education- SAMS Series.

3. Hebert Shildt: Programming in C# 4.0, Tata McGraw Hill.

ADDITIONAL LEARNING SOURCES:

1. https://msdn.microsoft.com/en-us/library/4w3ex9c2.aspx
2. http://www.asp.net/
3. http://www.aspfree.com/
4. http://www.devx.com/dotnet
5. asp.net-tutorials.com/localization/local-and-global-resources/
6. https://www.tutorialspoint.com/asp.net/asp.net_ado_net.htm
7. www.w3schools.com/asp/ado_intro.asp
8. https://www.tutorialspoint.com/soa/index.htm

CO-PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
CO	M	H	H	H	H	H	H	M	M	M	H	M
CO	H	M	H	M	M	M	H	L	M	M	M	L
CO	H	M	H	M	M	M	H	L	M	M	M	L
CO	H	M	M	M	M	M	H	L	M	M	M	L
CO	M	H	H	H	H	H	H	M	H	M	H	H

Department of Master of Computer Applications [MCA], SJCE, Mysuru							
Course Year	**Course Semester**	**Course Type**	**Credits**	**Contact Hours/ Week**			**Total Hours/ Semester**
				Theory	**Laboratory**	**Tutorials**	
III	**V**	**Theory**	**02**	**02**	**00**	**00**	**26**
Course No		**Course Title**				**Pre Requisites**	
MCA560		**Soft Skills-III : ENTREPRENUERSHIP**				**Soft Skills-I & III (Professional Communication, Effectiveness & Leadership)**	

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 3 Written Tests, 2-Events. Test] Total Marks: 50 [10* 5].
2. 2. Semester End Exam [Total Marks:50 Marks, Duration: 1.5 Hours]

COURSE OUTCOMES:

Upon successful completion of this course, students will be

CO 1: Understand the role of entrepreneur and importance of entrepreneurship in the current scenario.
CO 2: Identify the sources of idea generation and significance of feasibility of ideas.
CO 3: Understand the ownership structure and its appropriateness.
CO 4: Recognizing the funding institutions for entrepreneurship and examining the sources of venture capitals.
CO 5: Analyze the industry, market and operations of plan chosen and its impact o rural and social entrepreneurship.

TOPICS COVERED:

UNIT:1 – Foundation of Entrepreneurship **5 Hours**

Nature of entrepreneurship, social & cultural factors in nurturing Entrepreneurship. Institutional support for promoting entrepreneurship in India. Role of universities and colleges, CSIR labs in India to promote entrepreneurship.

UNIT:2 – Business Planning **5 Hours**

Idea generation to preparation of detailed plan. Project identification, selection, significance and feasibility.

UNIT:3 - Ownership Structures **5 Hours**

Proprietorship, partnership, company, co-operative. Selection of appropriate form of ownership structure.

UNIT:4 – Institutional Support to Entrepreneurs **5 Hours**

Need for institutional support. Institutional support to small entrepreneurs: NSIC, SIDO, SSIB, SSICS, SISI, DICs, Industrial estates, specialized institutions, TCOs.

Venture Capital: Valuing and financing a venture, stages of venture development and financing, venture expansion strategies.

UNIT:5 – Industry, Market and Operational Analysis **6 Hours**

Industry identification, trends and industry analysis. Analysis of market demand, supply, STP. Identifying operational process and product /service description.

Rural and social entrepreneurship: potential of entrepreneurship in rural India, SHGs, Micro credit facilities and problems of rural and social entrepreneurship.

TEXT BOOKS / REFERENCES:

Text books:

- Entrepreneurial Development: S S Khanka, First Edition, 2003, S Chand & Company .

Reference books:

- Entrepreneurship – Prof. T V Rao, Indian Edition 2009, S Chand & Company.

ADDITIONAL LEARNING SOURCES:

1. https://www.slideshare.net/esmatullahamini1/entrepreneurial-developmentbook-pdf.
2. www.roletech.net/books/Entrepreneurship2ndEdition.pdf.
3. www.unimaid.edu.ng/root/CDl/Entrepreneurship%20Studies.pdf.

CO-PO MAPPING:

	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO	PO1	PO1	PO12
CO	L	M	M	M	M	M	L	H	H	M	H	H
CO	L	H	H	M	L	M	M	H	M	M	M	H
CO	L	L	M	L	L	M	M	M	M	M	M	H
CO	L	L	L	L	L	M	L	L	M	M	M	H
CO	L	M	M	L	L	M	M	H	M	H	M	M

Department of Master of Computer Applications [MCA], SJCE, Mysuru.							
Course Year	**Course Semester**	**Course Type**	**Credits**	**Contact Hours/ Week**			**Total Hours/ Semester**
				Theory	**Laboratory**	**Tutorials**	
III	**V**	**Laboratory**	**01**	**00**	**03**	**00**	**48**
Course No		**Course Title**				**Pre Requisites**	

MCA51L	Python Programming Laboratory	OOPS

COURSE ASSESSMENT METHOD:

1. Internal Assessment [Test-1 + Test-2 + Observation book + Record + Viva: 15 + 15 + 05 + 05 +10] Marks: 50

COURSE OUTCOMES:

Upon successful completion of this course, students will be able to:

CO1: Understand and comprehend the basics of python programming.
CO2: Apply knowledge in real time applications.
CO3: Understands about files and its applications.
CO4: Apply object-oriented programming concepts to develop dynamic interactive Python applications
CO5: Demonstrate MVC architecture with respect to Django framework

LIST OF EXPERIMENTS COVERED:

1. Data Types and Data Structures :
Introduction to Python: - using the Python interpreter, Python built-in types, Arithmetic in Python, Program input and Program output, Variables and assignment. Strings and string operations, List basics, List operations, Dictionaries, Dictionary basics and Tuples,

(a) Simple programs using elementary data items, lists, dictionaries and tuples.

2. Control Structures:
Control Statements:-if statements, while statement, for statements, functions, formal arguments, variable-length arguments, Exceptions, detecting and handling exceptions.
(a) Programs using conditional branches, loops.
(b) Programs using functions
(c) Programs using exception handling

3. Classes ,files and modules
Introduction to Classes and Objects:-classes, class attributes, instances, instance attributes, binding and method invocation, inheritance, polymorphism, Built-in functions for classes and instances.
Files and input/output, reading and writing files, methods of file objects, using standard library functions, dates and times
(a) Programs using classes and objects
(b) Programs using inheritance
(c) Programs using polymorphism
(d) Programs to implement file operations.
(e) Programs using modules.

4. Database and web programming
Python database application programmer's interface (DB- API), connection and cursor objects, Type objects and constructors, python database adapters. Creating simple web clients, introduction to CGI, CGI module, building CGI applications, python web application frameworks: django.
(a) Programs using python database API.

(b) Programs for creating simple web pages.
(c) Programs for creating dynamic and interactive web pages using forms.

5. Development of sample web applications using python.
Sample applications may include
i) Web based polling
ii) Social networking site
iii) Online transaction system
iv) Content management system

TEXT BOOKS / REFERENCES:

Text books:

3. Timothy A. Budd: Exploring Python, Tata McGraw-Hill, 2011.
4. Jeff Forcier, Paul Bissex, Wesley Chun: Python Web Development with Django, Addison-Wesley (e-book), 2008.

Reference books:

1. Ascher, Lutz: Learning Python, 4th Edition, O'Reilly, 2009.
2. Wesley J Chun: Core Python Applications Programming, Pearson Education, 3rd Edition, 2013.
3. Paul Gries, Jennifer Campbell, Jason Montojo , Practical Programming: An introduction to Computer Science Using Python, second edition, Pragmatic Bookshelf.
4. Allen Downey , Jeffrey Elkner , Learning with Python: How to Think Like a Computer Scientist Paperback –, 2015

ADDITIONAL LEARNING SOURCES:

http://www.network-theory.co.uk/docs/pytut/
http://docs.python.org/tutorial/
http://zetcode.com/tutorials/pythontutorial/
http://www.sthurlow.com/python/ http://www.tutorialspoint.com/python/
http://www.djangoproject.com/
http://www.djangobook.com/

CO-PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO	PO9	PO1	PO1	PO12
CO1	H	L	L	M	L	L	L	L	M	L	L	L
CO2	L	M	M	M	H	L	L	L	M	H	L	L
CO3	H	H	H	H	M	M	L	L	L	L	L	L
CO4	H	H	H	H	M	M	L	L	L	L	L	L
CO5	L	H	L	L	H	M	L	L	L	L	L	L

Department of Master of Computer Applications [MCA], SJCE, Mysuru.							
Course Year	**Course Semester**	**Course Type**	**Credits**	**Contact Hours/ Week**			**Total Hours/ Semester**
				Theory	**Laboratory**	**Tutorials**	
III	**V**	**Laboratory**	**01**	**04**	**00**	**00**	**48**

Course No	**Course Title**	**Pre Requisites**
MCA52L	**Data Mining and Analytics Laboratory**	**Data Base Management Systems**

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: Test1 + Test2 + Observation book + Record + Viva-voce] Marks: 50 [15+15+10+5+5]

COURSE OUTCOMES:

Upon successful completion of this course, the student will be able to:
CO1: Interpret the basic concepts, principles and techniques of data mining.
CO2: Define knowledge discovery and data mining; recognize the key areas and issues in data mining.
CO3: To apply the techniques of clustering, classification, association finding, feature selection and visualization of real world data.
CO4: Understand and implement the basic concept and principles of data analytics.
CO5: Apply evaluation metrics to select appropriate techniques in data analytics.

LIST OF EXPERIMENTS COVERED:

The following are the list of algorithms where, the students are going to trace the steps of the algorithm using Advanced programming and create a front end and demonstrate the flow of the algorithm.

1. Frequent set identification using association rule.
2. Apriori Algorithm.
3. FP-Tree Algorithm.
4. K-mean Clustering Algorithm.
5. Agglomerative nesting clustering Algorithm.
6. Self Organizing Maps Algorithm (SOM).
7. DIC Algorithm.
8. Apriori Influencial Algorithm.
9. Partition Algorithm.
10. Page Rank Algorithm.
11. K-means Algorithm.
12. Birch Algorithm
13. ROCK Algorithm.
14. Finding distance interval scaled variables – Euclidian & Manhattan distance
15. Dissimilarity between binary variables –Symmetric/Assymmetric

TEXT BOOKS / REFERENCES:

1. Jiawei, Micheline Kamber, "Data Mining Concepts and Techniques", Morgan Kauf Mann Publishers.2012
2. Arun.K.Poojari, "Warehousing and Mining", PHI 2010.
3. Michael, J.Berry, Gordon Linoff: "Data Mining Techniques: Marketing, Sales, Customer support", John wiley & sons, 2010.

ADDITIONAL LEARNING SOURCES:

1.web.cse.ohio-state.edu/~srini/674/part1.ppt.
2.www.cse.iitb.ac.in/~dbms/Data/Talks/datamining-intro-IEP**.**

CO-PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	H	M	M	H	M	L	M	M	M
CO2	H	M	M	H	M	M	H	L	L	M	L	L
CO3	M	M	M	H	M	H	H	M	L	M	L	L
CO4	M	M	M	H	M	M	L	L	M	M	L	L
CO5	H	H	M	H	M	M	M	M	L	H	L	L

VI Semester

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Weeks/ Semester
				Theory	Laboratory	Tutorials	
III	VI	Project	25	00	00	00	18

Course No	Course Title	Prerequisites
MCA61P	Project Work	NIL

COURSE ASSESSMENT METHOD:

Internal Assessment [4 Events: Presentation and Report] Marks: 100.

COURSE OUTCOMES:

Upon Successful completion of the project, students are able to :

1. Apply the knowledge acquired within and beyond the curriculum.
2. Formulate and investigate a problem which is of current trend.
3. Apply Software Engineering methodologies by following Software Development Life Cycle (SDLC).
4. Communicate, interpret, design and implement efficiently.
5. Document and Present the processes involved in project implementation.

1. Introduction

Department of Computer Applications aims to impart value based technical education with right blend of computation skills, analyzing, designing and managing projects. The Project work constitutes a major component in most of the professional programmers. The project work not only aims at the partial fulfillment of the MCA requirements, but also provides a mechanism to demonstrate the student's skills, abilities and specialization of interest. As the program covers a wide range of theoretical and practical subjects related to computer applications, it is an opportunity for a student to showcase the combined practical implementation of all the courses covered in the earlier 5 semesters effectively and efficiently. The project work should compulsorily include the software development life cycle, documentation and demonstration of the same within the stipulated period of 18weeks.

1. Aim

The main aim of the project is to make students aware of the process of project implementation using their domain knowledge and to provide computer based solutions/applications for the industry, academic & research oriented real time problems.

2. Objective

The objective of the project is in such way that it helps the student in developing the ability to apply theoretical knowledge and practical tools/techniques.

3. Outcome

At the end of implementation of project, student is capable of,

- Understanding and applying Systems Development Life Cycle (SDLC).
- Identifying systems requirements.
- Applying different data gathering and analyzing techniques.
- Performing and evaluating feasibility studies like cost-benefit analysis, technical feasibility, time feasibility and operational feasibility for the project.
- Designing and developing of system architectures and data flow diagrams.
- Evaluating methods of process description to include structured, decision tables and decision trees.
- Creating and evaluating such alternative graphical tools as systems flow charts and state transition diagrams.
- Decide the software requirement specifications and hardware requirement specifications.
- Plan the systems design phase of the SDLC.
- Distinguish between logical and physical design requirements.

- Designing and evaluating system inputs and outputs.
- Estimate storage requirements.
- Explain the various file update processes based on the standard file organizations.
- Construct and evaluate Entity-Relationship (ER) diagrams for RDBMS related projects.
- Perform normalization for the un-normalized tables for RDBMS related projects.

4. Project Selection Process

Student can implement projects in two streams.

Application Stream

These projects aim at developing an application for attaining a specific or group of tasks in respective domains. The student has to understand the applicability of the project thoroughly and should develop application based on the required features and functionalities. The following are some important application domains. These projects could be done in college or industry. Student can take up any one the following application domains, but not limited to these:

- Database, Data mining and Data warehousing applications
- Networking applications [Mobile, SAN, Security, MANet etc…]
- Business / Enterprise applications [Supply Chain, ERP, CRM etc…]
- Web based applications
- Multimedia applications
- Software Engineering applications
- Grid and Cloud computing applications
- Parallel and distributed computing applications
- E-learning and e-commerce applications
- Gaming applications
- Hardware driver applications
- Mobile applications
- Healthcare applications
- Banking and Finance applications
- Insurance applications

Research Stream

The problem formulation should made it clear to everyone, students should aim to address whom and where it is relevant. Chosen areas can be any of the following or any area of research.

- Public health, Epidemiology & Health Services
- Bio medical.
- Clinical sciences and health practice.
- Neurosciences & Behavioral sciences.
- Cloud services
- Big data
- Data science

- Image processing
- Internet(IOT)
- Any engineering topics where software plays vital role.

One of the most important early step in a research projects is the literature survey. A literature survey is done to identify related research done already and to set the current research work within a conceptual and theoretical context.

5. Problem Formulation

Students should follow the following steps (Respective guides to encourage students to follow these steps while formulating the problem for the project).

- Tailoring the idea to the targeted topic of the work programme.
- Describe the objectives and applications of the project – setting the specific objectives of the topic.
- Set up the project implementation phases and structure of the project – what activities you need to achieve the objectives and what expertise/knowledge is needed.
- Describe the end results (project outputs) and the impact of the project outcome.
- Highlight the added value, the innovative element of your project.
- Chosen project concept should address the main questions: WHY? WHAT? HOW? WHO? WHERE?

o Statement of Problem: the "Why?"

Summary of the original problem statement.

Background: Brief description of company and their business relevance or importance of problem. Background information to educate the reader previous related work by others—literature review with credible sources patent search on related problem.

o Objectives: the "What?"

In the Objectives section, student should translate the customer's quantitative and qualitative needs into clear, objective design specifications. Define the scope of work and clearly state the project objectives.

o Technical Approach: the "How?"

Although you may not know all the details of the problem solution, you should know a first design on how you will solve the problem, and you should have some design concepts.

o Intended users: Who?

This should address who is the user of the system/beneficiary of the work.

o Environment: Where?

This should address area of domain, your application or work will be useful.

6. Evaluation Process

The project duration is 18 weeks; 16 weeks for development and 2 weeks for preparation of the project report. Students needs to maintain the project dairy, consult their internal guide and update the status of the work and get it reviewed every week. Students are be required to provide their project implementation details, methodology and results to the department. otherwise students will forfeit the claim for having done the project work.

Evaluation process has following steps.

Project Screening and Finalization

Student activity: Submission synopsis and finalization.

2) **Evaluation of system study and analysis**

Student activity: Submission of system requirement specification

Evaluation of project design

Student activity: Submission of design document

4) **Evaluation of coding, testing and implementation.**

Student activity: Submission of draft final project report

5) **Final report submission.**

Project screening

Student projects will be screened based on following points

a) Relevance to current technology
b) Gauging whether project work involves 16 week man hours of work.
c) Place of execution: Industry/Research organization/College
d) Project relevance to Computer Science and Applications
e) Understanding of the project work by the student

Format of the Synopsis can be referred in **Annexure-I**

Finalization of Synopsis

a) Finalization of Synopsis is done based on the previous stage of evaluation
b) Based on recommendation of the panel comprising of all the staff members of the department.
c) Verifying whether suggestions of the previous evaluation incorporated or not.

Format of the Synopsis can be referred in **Annexure – I**

Evaluation of system study and requirement analysis

Evaluation criteria in this step as follows.

a) Whether student understood the problem in depth.
b) Whether scope and objective of the project defined properly.
c) Whether student studied and understands the system thoroughly/completely.
d) Whether functional requirements and non functional requirements defined properly.

SRS document can be referred in **Annexure-II**

Evaluation of project design

In this level evaluated for

a) Architectural Design
b) Detailed Design
c) Database Design
d) External Interface Design
e) Algorithm Design

Design document format as per in **Annexure-III**

Evaluation of coding, testing and implementation

Evaluation of coding, testing and implementation criteria as follows.

a) Pseudo code
b) Test case generation
c) Actual Testing
d) Users Manual
e) Implementation strategy
f) Draft report submission

g) Checking whether the implementation has meet requirements specified in SRS document or not.
h) Demo of the final working model of the system.

Final report format should be followed according **Annexure-VI**

Annexure I: Format of the Synopsis

Application stream

- Title of the Project.
- Problem statement.
- Objective.
- Scope of the project.
- Brief description.
- Project Category (RDBMS/OOPS/Networking/E-learning.,etc.).
- Software requirements.
- Hardware requirements

Research stream

- Title of the Project.
- Problem statement.
- Objective of the project.
- Related work done(list of literature)
- Brief description of proposed work.
- Software requirements.

Note :
Students doing internship program outside the college need to submit the following details.

- Name and address of the company
- External guide name and designation
- Email and contact number of the external guide.

Annexure II: Format of the SRS document

Application stream

Introduction:

Scope of SRS document
Definitions, acronyms and abbreviations
- Definitions, acronyms and abbreviations used.

References.
Overview.

Overall description:

Product perspective
- About your project work.

Product functions
User characteristics
- Characteristics of the intended user.

General constraints
Assumptions and Dependencies

Specific requirements:

External Requirements
(User Interface/Hardware/Software)
Functional Requirements
Non functional requirements
Other requirements.

Architectural overview
Data flow diagrams
Design constraints

Research stream

Introduction.
Detailed literature review of related work.
Detailed analysis of the proposal work
Conclusion.

Annexure III: Format of Design Document:

Application stream

Introduction

Scope of design document
Definitions, acronyms and abbreviations

References
Overview
System Architecture
Architectural design
User interface design
Forms design
Database or file design
Reports design
Workflows and algorithms design
Detailed design of DFDs.
Activity diagrams
Use case diagrams
Sequence diagrams
Component interfaces
Component design description

Software requirements mappingCritical requiremnts

Research stream:

Introduction
Algorithms/logic used in the related work.
Algorithms/logic steps involed in the proposed method.
Conclusion

Annexure IV : Coding detail

Application and Research stream:
Introduction
Coding convention.
Pseudo code of the important algorithms of the project.

Annexure V: Testing report
Application and Research stream:

Introduction
Test process done for the project.
Test cases.

Testcase no	Positive scenario	Required input	Expected output	Actual output	Test pass/fail

Testcase no	Negative scenario	Required input	Expected output	Actual output	Test pass/fail

Annexure VI : Report format.

TITLE PAGE

CERTFICATE by the Department

CERTIFICATE by the EXTERNAL ORGANIZATION

DECLARATION

ACKNOWLEDGEMENTS

ABSTRACT

CONTENT PAGE

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1:

Introduction to the Topic of the Project

Problem definition and Premises

Objectives of the Study / Project

Methodology

Organization of the Dissertation

CHAPTER 2:

History of the Organization (If carried out in an organization)

CHAPTER 3:

Justification for doing this project

CHAPTER 4:

Requirements analysis and specifications document. Refer Annexure II

CHAPTER 5:
Design Document
Refer Annexure III

CHAPTER 6:

Coding

Refer Annexure IV

CHAPTER 7:

Testing

Refer Annexure V

CHAPTER 8:

Experimental results/findings.

Conclusion

Scope for future work (If any)

Limitations of the study (If any)

References [Papers, books, thesis web references (urls)]

List of Publications (if any)

Attach hard copy of the publication.

Note:

- Page numbers from “Abstract” to “List of Tables” to be given in Roman Numerals
- Page numbers to be given in numerals from Chapter 1 to Chapter 7.
- All pages should contain a footer indication the Department of Master of Computer Applications, SJCE, Mysuru and header indicated the title of the project and Month-Year of submission.

Font size should be:

- 16 bold for chapter no and chapter title.
- 14 bold times new roman for sub headings.
- 12 bold times new roman for sub-sub headings.
- 12 times new roman normal for running text.
- Table numbers, figure numbers and photograph numbers should be according to Chapter with appropriate title. Caption should contain figure no and title below the figure. Table no and title should be above the table.
- Spacing should be 1.5 between lines and 2.0 between paragraphs.
- Cross references should be clearly indicated and proper citation should be given Wherever references have been made.
- Project report should be hard bound in Pink color to be submitted. Make 3 Copies. Submit one to Department, one to internal guide and one to yourself.

Marks allotment in evaluation process

The evaluation process is done in four stages.

First Evaluation: Evaluation is conducted at the time of acceptance of the project i.e., during synopsis submission. Students needs to prepare and present the selected project before the panel. Evaluation carries 20 marks. The main components to be evaluated are,

Sl. No	Evaluation of Synopsis	Panel Marks	Guide Marks
1	Role of Innovation / Creativity, Research Element, challenges in the Problem.	2	2
2	Literature survey / Existing system analysis.	2	2
3	Clarity about the problem	2	2
4	Attendance	2	2
5	Question and Answer	2	2

Note:

If total marks is less than 50% of the maximum mark, then after a week student need to re-present the project before the panel for reconsideration of the project or may propose a different project. After the panel clearance students can start the project.

Second Evaluation: Evaluation is conducted after one month of submission of the synopsis. Students needs to prepare and present on the methodology, analysis and possible outcome of the problem in front of the panel. Evaluation carries 20 marks. The main components to be evaluated are, understanding of the problem, system study and analysis done.

Sl.No	Evaluation of SRS	Panel Marks	Guide Marks
1	Analysis of the Problem and understanding	2	2
2	Specification of the requirements	2	2
3	Progress of the project	2	2
4	Attendance	2	2
5	Question and Answer	2	2

Third Evaluation:

Evaluation is conducted after one month of second evaluation.. Students needs to present the design concepts of the project in front of the panel. Evaluation carries 20 marks. The main components to be evaluated are,

Sl.No	Evaluation of SRS	Panel Marks	Guide Marks
1	Architectural Design	2	2
2	Detailed Design	2	2
3	Work implementation status	2	2
4	Attendance	2	2
5	Question and Answer	2	2

Final Evaluation:

Evaluation is conducted after one month of third evaluation to check coding, testing and implementation stages. Students need to prepare draft final report with demo of the project for the presentation.

Sl.No	Evaluation of SRS	Panel Marks	Guide Marks
1	Coding and implementation efficiency	4	4
2	Work completion	4	4
3	Testing efficiency	4	4
4	Attendance	4	4
5	Question and Answer	4	4

Overall marks distribution:

Total internal assessment marks	Panel member	Guide
100	**50**	**50**

Total marks distribution:

Project	Internal Assessment	Report Evaluation	Report Evaluation	Viva Voce	Total Marks

	(Guide (50%) + Panel (50%))	Internal (I)	External (E)	Avg. (Internal and External)		
Major	100	100	100	100	50	250
Mini	30	10	----	10	10	50

Credits distribution for the evaluation for major project.

Evaluation Steps	Credits
First evaluation	**2**
Second evaluation	**2**
Third evaluation	**2**
Final evaluation	**4**
Report Evaluation.{Average of internal and external examiners}	**10**
Viva Voice	**5**
Total	**25**

Note:

1) Mini project will be evaluated for 100 marks and reduced to 50 marks at the end .

2) Mini project will be for 2 credits.