JSS Mahavidyapeetha

JSS Science and Technology University
Mysuru - 570 006.

Department of Master of Computer Applications

OVERALL COURSE GRID OF

Master of Computer Applications

From I to VI Semesters

2018-2021

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	Java Programming	MCA	4	0	0	4.0	4	50	50	100	3
2.	MCA120	Data Structures using C	MCA	4	0	0	4.0	4	50	50	100	3
3.	MCA130	Management Information System	MCA	4	0	0	4.0	4	50	50	100	3
4.	MCA140	System Software	MCA	4	0	0	4.0	4	50	50	100	3
5.	MCA150	Mathematical foundations	MCA	4	0	0	4.0	4	50	50	100	3
6.	MCA11L	Java Programming Laboratory	MCA	0	0	1	1.0	3	50	00	50	0
7.	MCA12L	Data Structures using C Laboratory	MCA	0	0	1	1.0	3	50	00	50	0
8.	MCA14L	System Software Laboratory	MCA	0	0	1	1.0	3	50	00	50	0
9.	MCA15L	Mathematical foundations Laboratory	MCA	0	0	1	1.0	3	50	00	50	0
Total				20	0	4	24	32	450	250	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	Advanced Java Programming	MCA	4	0	0	4.0	4	50	50	100	3
2.	MCA220	Analysis and Design of Algorithms using C++	MCA	4	0	0	4.0	4	50	50	100	3
3.	MCA230	Software Engineering	MCA	4	0	0	4.0	4	50	50	100	3
4.	MCA240	Operating System with Linux	MCA	4	0	0	4.0	4	50	50	100	3
5.	MCA250	Relational Database Management Systems	MCA	4	0	0	4.0	4	50	50	100	3
6.	MCA21L	Advanced Java Programming Laboratory	MCA	0	0	1	1	3	50	00	50	-
7.	MCA22L	Analysis & Design of Algorithmsusing C++ Laboratory	MCA	0	0	1	1	3	50	00	50	-
8.	MCA23L	Software Engineering Laboratory	MCA	0	0	1	1	3	50	00	50	-
9.	MCA24L	Operating System and Linux Laboratory	MCA	0	0	1	1	3	50	00	50	-
10.	MCA25L	Relational Database Management Systems Laboratory	MCA	0	0	1	1	3	50	00	50	-
Total				20	0	5	25	35	500	250	750	-

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	Web Programming	MCA	4	0	0	4.0	4	50	50	100	3
2.	MCA320	Intelligent Algorithms	MCA	4	0	0	4.0	4	50	50	100	3
3.	MCA330	Object Oriented Modeling and Design	MCA	4	0	0	4.0	4	50	50	100	3
4.	MCA340	Computer Networks	MCA	4	0	0	4.0	4	50	50	100	3
5.	MCA35X	Streams	MCA	4	0	0	4.0	4	50	50	100	3
6.	MCA31L	Web Programming Laboratory	MCA	0	0	1	1	3	50	-	50	-
7.	MCA32L	Intelligent Algorithms Laboratory	MCA	0	0	1	1	3	50	-	50	-
8.	MCA33L	Object Oriented Modelingand Design Laboratory	MCA	0	0	1	1	3	50	-	50	-
9.	MCA34L	Computer Networks Laboratory	MCA	0	0	1	1	3	50	-	50	-
Total				20	0	4	24	32	450	250	700	-

Streams :

Stream-1		Stream-2		Stream-3	
Machine Intelligence Stream		**Data Sciences Stream**		**Applications Stream**	
MCA351	Computer Vision and Pattern Recognition	MCA352	Data Warehouse	MCA353	Enterprise Resource Planning

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	Python Programming	MCA	4	0	0	4.0	4	50	50	100	3
2.	MCA420	Data and Web Mining	MCA	4	0	0	4.0	4	50	50	100	3
3.	MCA430	Software Architecture	MCA	4	0	0	4.0	4	50	50	100	3
4.	MCA440	Cryptography and Network Security	MCA	4	0	0	4.0	4	50	50	100	3
5.	MCA45X	Streams	MCA	4	0	0	4.0	4	50	50	100	3
6.	MCA41L	Python Programming Laboratory	MCA	0	0	1	1.0	3	50	-	50	-
7.	MCA42L	Data and Web Mining Laboratory	MCA	0	0	1	1.0	3	50	-	50	-
8.	MCA43L	Applications Laboratory	MCA	0	0	1	1.0	3	50	-	50	-
9.	MCA44L	Cryptography and Network Security Laboratory	MCA	0	0	1	1.0	3	50	-	50	-
Total				20	0	4	24	32	450	250	700	-

Streams :

Stream-1		Stream-2		Stream-3	
Machine Intelligence Stream		**Data Sciences Stream**		**Applications Stream**	
MCA451	Artificial Intelligence and Robotics	MCA452	NoSQL	MCA453	Business Intelligence

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	.NET Technologies	MCA	4	0	0	4.0	4	50	50	100	3
2.	MCA520	Mobile Applications	MCA	4	0	0	4.0	4	50	50	100	3
3.	MCA530	Software Testing and Practices	MCA	4	0	0	4.0	4	50	50	100	3
4.	MCA540	Cloud Computing	MCA	4	0	0	4.0	4	50	50	100	3
5.	MCA55X	Streams	MCA	4	0	0	4.0	4	50	50	100	3
6.	MCA51L	.NET Technologies Laboratory	MCA	0	0	1	1.0	3	50	-	50	-
7.	MCA52L	Mobile Applications Laboratory	MCA	0	0	1	1.0	3	50	-	50	-
8.	MCA54L	Current Trends in IT Laboratory (Exploration and Presentation)	MCA	0	0	1	1.0	3	50	-	50	-
Total				20	0	3	23	29	400	250	650	-

Streams :

Stream-1		Stream-2		Stream-3	
Machine Intelligence Stream		**Data Sciences Stream**		**Applications Stream**	
MCA551	SoftComputing	MCA552	Big Data Analytics	MCA553	Software Project Management and Practices

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	15	15.0	3	100	150	250	1
Total				0	0	15	15.0	3	100	150	250	-

Credit Pattern for MCA Course

Semester	Credits
1	24
2	25
3	24
4	24
5	23
6	15
Total	135

I Semester

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**	
I	I	Theory	04	04	00	00	52
Course No		**Course Title**				**Pre Requisites**	
MCA110		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 able to:

CO1: Demonstrate and implement programs using components and constructs of a Java language
CO2: Identify classes, objects, members of a class and use packages and interfaces appropriately.
CO3: Demonstrate for Java program for multithread, synchronization and exception handling concepts.
CO4: Use the concept of string, event handling, simple data structures like arrays and members of classes of Java API in application development
CO5: Design and develop Java based UI and Networking applications using applets, swing components and networking concepts.

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 and 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:

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

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	
I	I	Theory	04	04	00	00	52

Course No	Course Title	Prerequisites
MCA120	Data Structures using C	--

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, the student will be able to:

CO1: Design and analyze programming problem statements.
CO2: Choose appropriate data structures and algorithms, understand the ADT/libraries, and use it to design algorithms for a specific problem.
CO3: Apply mathematical abstraction to solve problems.
CO4: Demonstrate various methods of organizing large amounts of data.
CO5: Analyze algorithms and to determine algorithm correctness and time efficiency class.

TOPICS COVERED:

UNIT 1 - Introduction and overview of C Programming 12 Hours

Introduction to C programming, Variables, Data types, Constants, Declarations, Operators, Precedence, Associativity, Order of evaluation. Input and output statements; Control Statements, Arrays – Single dimension, Two dimensional, Multi dimensional Arrays, Strings. Functions, Categories of functions. Examples Pointers, Pointer arithmetic, Call by value, Pointer Expression, Pointer as function arguments, recursion, passing strings to functions, Call by reference, Functions returning pointers, Pointers to functions, Programming Examples. Structures and Unions.

UNIT 2 - Introduction to Data Structures 8 Hours

Information and its meaning: Abstract Data Types, Sequences as Value Definitions, ADT for Varying length character Strings, pointers and review of Pointers, Dynamic Memory Allocation - definition, malloc, calloc, and realloc, free. Data Structures: Array as an ADT, 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 12 Hours

The queue and its sequential representation, the queue as ADT, Basic operations 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 and Trees 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. Tree: Definition and representation,

Types of trees, Basic operations on Tree.

TEXT BOOKS/ REFERENCES :

Text Books :

1.Programming in ANSI C, Third Edition, E. Balaguruswamy. 6th Edition (2013).

2. Data Structures Using C and C++ by Aaron.M. Tenenbaum, Yedidyah Langsam and Moshe J. Augustine, PHI, Edition, 2011.

Reference Books :

1. Data structures, Algorithms and Applications in C++, S. Sahani, University Press (India) Pvt Ltd, 2nd Edition.
2. The complete reference C, Herbert Schildt, Fifth Edition, Tata McGraw Hill.

ADDITIONAL LEARNING SOURCES:

1. http://www.tutorialspoint.com/Data-Structures-in-C-Online-Training/classid=13
2. http://nptel.ac.in/datastructures_c

CO - PO MAPPING:

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

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	
I	I	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA130	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, the student will be able to:

CO1: Apply the different strategies for the management of business to formulate business process.
CO2: Analyze the need for business process re-engineering, and the process of making.
CO3: Analyze and examine business information needs to facilitate evaluation of strategic alternatives.
CO4: Apply Management Information Systems knowledge and skills learned to facilitate the acquisition, development, deployment, and management of information systems.
CO5: Effectively communicate strategic alternatives to facilitate decision-making.

TOPICS COVERED:

UNIT 1- Systems Engineering, Information and Knowledge 12 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,
Introduction of MIS: MIS: Concept, Definition, Role of the MIS, Impact of MIS, MIS and the user, Management as a control system, MIS support to the management, Management effectiveness and MIS, Organization as system. MIS: organization effectiveness.

UNIT 2- Decision Making and DSS 10 Hours

Decision making concepts; decision making process, decision-making by analytical modeling, and Behavioral concepts in decision making, organizational decision-making, Decision structure, DSS components, and Management reporting alternatives
Technology of Information System : 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 3- Electronic Business systems 10 Hours

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 (real world cases for the above)
E-business Technology: 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.

UNIT 4- Strategic Management of Business & Developing Business/IT Strategies /IT

Solutions **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 5: E-Commerce Introduction **10 Hours**

Course overview; Introduction to e-commerce, E-commerce Business Models and Concepts, Ecommerce Infrastructure: The Internet and World Wide Web, Web design, JavaScript Internet Information Server (IIS); Personal Web Server (PWS).

E-Commerce techniques and Issues: Introduction to Active Server Pages (ASP), Building an E-Commerce Web Site, E-Commerce Payment Systems, E-Commerce Marketing Techniques, Building product catalogue, Search product catalogue, Web Spider and search agent, Ethical, Social and Political Issues in E-Commerce.

TEXT BOOKS /REFERENCES:

Text Books :

1. Waman S Jhawadekar: Management Information System, 3rd Edition, Tata McGraw Hill.
2. James A O'Brien and George M Marakas: Management Information System, 7th Edition, Tata McGraw Hill, 2006
3. Turban, Rainer, and Potter, Introduction to E-Commerce, second edition, 2003
4. H. M. Deitel, P. J. Deitel and T. R. Nieto, E-Business and E-Commerce: How to Program, Prentice hall, 2001

Reference Books:

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, 2001
3. Elizabth Chang: E-Commerce Fundamentals and Applications, Wile India Edition.

ADDITIONAL LEARNING RESOURCES:

1. http://mbaexamnotes.com/management-information-system-notes
2. https://www.tutorialspoint.com/management_information_system

CO-PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	M	H	M	M	M	H	H	H	M	M	L

CO2	H	H	M	H	L	H	M	H	H	M	H	M
CO	M	H	M	M	H	H	H	M	L	H	H	M
CO4	M	M	L	H	M	L	M	M	L	H	H	H
CO5	L	L	M	M	H	L	L	M	M	L	M	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	
I	I	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA140	System Software	-

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 to:

CO1: Compare and Categorize SIC & SIC/XE Machine architectures and working of assemblers, design Principles
CO2: Analyze and Determine loader and linker functionalities, structure of editors and interactive debugging systems
CO3: Summarize the working , design and structures of a macro processor
CO4: Demonstrate the working of compilers and various design options
CO5: Design the fundamentals of scanner and parser and implement the lexer and parser using LEX and YACC tools.

TOPICS COVERED:

UNIT 1 -Machine Architectures and Assemblers **14 Hours**

Introduction, System Software and Machine Architecture, Simplified Instructional Computer (SIC) - SIC Machine Architecture, SIC/XE Machine Architecture, SIC and SIC/XE ,Examples.
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 2 - Loaders, Linkers, Editors and Debugging Systems **12 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 3 -Macro Processor **8 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 4 - Compilers **6 Hours**

Compilers- Basic compiler functions: grammers, lexical analysis, syntactic analysis, code generation. Design options: division into passes, interpreters, p-code compliers and complier-compliers, implementation examples: Sun0s C Complier and The Yacc complier-complier.

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.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:

Text books:

1. Leland. L.Beck, D Manjula: System Software, 3rd Edition, Pearson Education, 2013.

2. John.R.Levine, Tony Mason and Doug Brown: Lex and Yacc, O'Reilly, SPD, 2009.

Reference books:

1. D.M.Dhamdhere: System Programming and Operating Systems, 2nd Edition, Tata McGraw – Hill, 2010.

ADDITIONAL LEARNING SOURCES:

1. http://techterms.com/definition/systemsoftware.
2. Studymca.blogspot.com/2009/01/system-software_24.html.
3. cse.iitkgp.ac.in/~bivasm/notes/LexAndYaccTutorial.pdf

CO-PO MAPPING:

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

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	
I	I	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA150	Mathematical Foundations	----------

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: Implement statistical measures and explore its applications
CO2: Analysis of computational errors and design of algorithms to solve a set of linear equations.
CO3: Applying the concepts of vector and linear functions in real time applications.
CO4: Apply the notion of relations on finite structures, like strings and analyze algorithms
using the concept of functions.
CO5: Explore the properties of Graph theory and its applications in computer science.

TOPICS COVERED:

UNIT 1- Statistics **10 hours**

Univariate data – different measures of location, dispersion, relative dispersion, skewness and kurtosis, Moments, Measures based on them – comparison with moment measures, Correlation and Regression Analysis.

UNIT 2 – Number Systems and Vector & Matrix Algebra **10 hours**

Errors in Numerical Computations, Types of Errors, Analysis and Estimation of Errors, Vector Algebra: Vector spaces with real field, Basis and dimension of a vector space, Orthogonal vectors, Properties of Matrices and Determinants: Matrix Operations, Elementary Matrices,

Inverse Matrix, Diagonal Matrix, Symmetric Matrix, and Determinant Matrix.

UNIT 3 - Linear Algebraic Systems **11 hours**

Numerical methods for Linear Systems, Direct Methods for Linear Systems: Cramer's Rule, Gauss Elimination Method, Gauss Jordan Elimination Method, Pivoting Strategies, Gauss-Jordon Method, LU Decomposition Method, Tridiagonal Systems of Linear Equations, Iterative Methods for Solving Linear Systems, Jacobis Iteration Method, Gauss-Seidel Iterative Method, Convergence Criteria, Eigen Values and Eigen Vectors.

UNIT 4 – Relations and Functions **10 hours**

Cartesian products and Relations, Properties of 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.

UNIT 5 - Graph Theory **11 Hours**

Definitions and Examples, Subgraphs, Complements, and Graph Isomorphism, Vertex Degree: Euler Trails and Circuits, Planar Graphs, Hamilton Paths and Cycles, Graph Coloring and Chromatic Polynomials.

TEXT BOOKS / REFERENCES:

TEXT BOOKS :

1. Sant Sharan Mishra, "Computer Oriented Numerical and Statistical Methods", PHI Learning Private Limited, 2013.
2. Rizwan Butt, "Introduction to Numerical Analysis Using Matlab", Infinity Science Press LLC, 2008
3. Ralph P Grimaldi, B.V.Ramana, "Discrete & Combinatorial Mathematics, An Applied Introduction" 5th Edition, Pearson Education, 2009.

REFERENCES:

1. D.S. Chandrasekharaiah, Discrete Mathematical Structures, 4th Edition, PRISM Pvt. Ltd. 2012.
2. Bondy and U.S.R.Murty: Graph Theory and Applications (Freely downloadable from Bondy's website; Google-Bondy)
3. S. Kumarsean, "Linear Algebra A geometric approach", Prentice Hall of India Private Limited, 2001.
4. Kenneth H Rosen, "Discrete Mathematics & its Applications" 7th edition, McGraw-Hill, 2010.

ADDITIONAL LEARNING SOURCES:

1. http://www.personal.kent.edu/~rmuhamma/GraphTheory/graphTheory.htm

2. http://www.tutorialspoint.com/discrete_mathematics/
3. http://nptel.iitm.ac.in/
4. http://www.maths.lu.se/english/library/e-resources/
5. http://sunzi.lib.hku.hk/ER/detail/hkul/3743848
6. https://www.**math**.ucdavis.edu/~**linear**/**linear**-guest.pd

CO-PO MAPPING:

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

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	
I	I	Laboratory	01	00	03	00	48

Course No	Course Title	Pre Requisites
MCA11L	Java Programming Laboratory	--

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: Test1+Test2 + Observation book+ Record + Viva-voce] Marks: 50 [15+15+5+5+10]

COURSE OUTCOMES:

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

CO1: Demonstrate and implement programs using components and constructs of a Java language

CO2: Identify classes, objects, members of a class and exhibit use packages and interfaces appropriately.

CO3: Demonstrate for Java program for multithread, synchronization and exception handling concepts.

CO4: Use the concept of string, event handling, simple data structures like arrays and members of classes of Java API in application development

CO5: Design and develop Java based UI and Networking applications using applets, swing components and networking concepts.

LIST OF PROGRAMS TO BE COVERED:

1. Display Hello world
2. Check entered number is ODD or EVEN
3. Find factorial of number
4. Find the sum of the digits of a given number
5. Swap two numbers without using a temporary variable
6. Accept a name and display the name with greeting message using Class.
7. 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. 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. Demonstrate Constructor Overloading and Method Overloading.
10. 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. 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

Consumer concept.

16. Write a JAVA program to implement a Queue using user defined Exception Handling (also make use of throw, throws.).
17. 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.
18. 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.
19. 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.
20. Print a chessboard pattern.
21. Write a JAVA Program which uses FileInputStream / FileOutPutStream Classes.
22. Demonstrate utilities of LinkedList Class.
23. Write a JAVA applet program, which handles keyboard event.
24. Write a JAVA Swing program, to design a form.
25. Create a simple Student_Registaration application using Swings, JDBC and MySQL.
26. Write a JAVA program which uses Datagram Socket for Client Server Communication.

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:

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

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	
I	I	Laboratory	01	00	03	00	48

Course No	Course Title	Prerequisites
MCA12L	Data Structures using C Laboratory	--

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:

CO1: Design and analyze programming problem statements.
CO2: Choose appropriate data structures and algorithms, understand the ADT/libraries, and use it to design algorithms for a specific problem.
CO3: Apply mathematical abstraction to solve problems.
CO4: Demonstrate various methods of organizing large amounts of data.
CO5: Analyze algorithms and to determine algorithm correctness and time efficiency class.

LIST OF EXPERIMENTS COVERED:

PROGRAMS ON C.

1. Programs to learn and explore C data types, looping and decision making structures.

{mean, median, lcm, gcd, min max}

2. 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.

3. Solve quadratic equations to find the roots of the equation.

4. Programs to implement arrays and structures.{Ex: Students marks calculation, matrix operations}

5. Calculate the average marks of the student test marks and display the result using structure.

6. Programs to implement dynamic memory allocation: malloc, calloc, realloc and free.

STACK

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.
10. Write a C program to implement multiple stack of integers.

QUEUES

11. Write a C program to perform basic operations on queue of integers, 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 operations using linked list.
15. Write a C program to implement queue operations 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.
17. Write a C program to perform operations.
a. Creation of list.
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.
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 :

Text Books :

1. Programming in ANSI C, Third Edition, E. Balaguruswamy. 6th Edition (2013).
2. Data Structures Using C and C++ by Aaron.M. Tenenbaum, Yedidyah Langsam and Moshe J. Augustine , PHI, Edition, 2011.

Reference Books :

1. Data structures, Algorithms and Applications in C++, S. Sahani, University Press (India) Pvt Ltd, 2nd Edition.
2. The complete reference C, Herbert Schildt, Fifth Edition, Tata McGraw Hill.

ADDITIONAL LEARNING SOURCES:

1. http://www.tutorialspoint.com/Data-Structures-in-C -Online-Training/classid=13
2.http://nptel.ac.in/datastructures_c

CO - PO MAPPING:

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

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	
I	I	Lab	01	00	03	00	48

Course No	Course Title	Pre Requisites
MCA14L	System Software Laboratory	-

COURSE ASSESSMENT METHOD:

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

COURSE OUTCOMES:

Upon successful completion of this course, students will be

CO1: Investigate and summarize the need and importance of Lex and Yacc tools.
CO2: Analyze and Construct the usage of different regular expressions

CO3: Demonstrate the structure and working principle of Lex and Yacc programs.
CO4: Construct and Practice different Lex programs**.**
CO5: Construct and Practice different Yacc programs**.**

TOPICS 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

1. To test the validity of a simple expression involving operators +,-,*,/.
2. To test the simple validity of postfix expression.
3. To recognize nested IF control statements and display the number of levels of nesting.
4. To recognize nested FOR control statements and display the number of levels of nesting.
5. To recognize nested WHILE control statements and display the number of levels of nesting.
6. To test the valid arithmetic expression that uses operators +,-,* and /.
7. To recognize a valid variable, which starts with a letter, followed by any number of letters or digits.
8. To recognize a valid variable , which starts with a letter with underscore, followed by any number of letters or digits.
9. To recognize a valid variable, which starts with a letter, followed by any number of letters , digits and special symbols' present ina file..
10. To evaluate an arithmetic expression involving operators +,-,* and /:
11. To recognize strings 'aaab', 'abbb', 'ab' and 'a' using the grammar ($a^n b^n$, n>=0)

12. To recognize the grammar (a^nb, n>=10)
13. To demonstrate what Yacc cannot parse.

TEXT BOOKS / REFERENCES:

Text books:

1. John.R.Levine, Tony Mason and Doug Brown: Lex and Yacc, O'Reilly, SPD, 1999.

Reference books:

1. John.R.Levine: Flex and Bison, O'Reilly, SPD, 2012

ADDITIONAL LEARNING SOURCES:

1. http://dinosaur.compilertools.net/
2. epaperpress.com/lexandyacc/download/LexAndYaccTutorial.pdf
3. ssd.sscc.ru/old/school/2005/files/nechaev.ppt

CO-PO MAPPING:

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

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	
I	I	Laboratory	01	00	03	00	48

Course No	Course Title	Pre Requisites
MCA15L	Mathematical Foundations Laboratory	--------

COURSE ASSESSMENT METHOD:

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

COURSE OUTCOMES:

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

CO1: Implement statistical measures and explore its applications
CO2: Analysis of computational errors and design of algorithms to solve a set of linear equations.
CO3: Applying the concepts of vector and linear functions in real time applications.
CO4: Apply the notion of relations on finite structures, like strings and analyze algorithms
using the concept of functions.
CO5: Explore the properties of Graph theory and its applications in computer science.

TOPICS COVERED:

Students are instructed to implement the algorithms using Scilab / Matlab tool.
1. To perform data analysis and modeling by writing the program to find I order, II order and III order statistical measure of varied forms of data set and understand its inference on the applications of data.
[Write program using adequate data to infer the meaning of Mean, Median, Mode, SD, Variance, Kurtosis, Correlation and Regression Analysis]

2. Analyze different types of Errors and its sources. Explore the implementation of demonstrate the concepts in linear algebra: Viz Matrices, Vector space representation, Polynomials considering the real time data using Matlab.
[Program to find the Inverse, Determinants of a matrix, Finding the Eigen value , Eigen Vector, Singular Value decomposition of a given matrix]

3. To implement the algorithms used to solve a set of linear equations (atleast any 3) and analyse the computational efficiency.
[Write Program and cross validate the results using the tools to solve the set of linear equations using : Gauss Elimination, LU Decomposition, Gauss Jordon, Jacobis Methods]

4. Implement the concepts of functions and relations.
[Program to validate Relations and Functions, its different types. generate Stirling Numbers, The Pigeon-hole principle considering the appropriate data for input and conclude the hypothesis involved in them to infer the results]

5. Implement to find the Hamiltonian, Euler, and isomorphic graphs.

What does Scilab / Matlab / C do ?
Includes hundreds of mathematical functions. It has a high level programming language allowing access to advanced data structures, 2-D and 3-D graphical functions.

A large number of functionalities is included in Scilab:
For usual engineering and science applications including mathematical operations and data analysis. 2-D&3-DVisualization graphics functions to visualize, annotate and export data and many ways to create and customize various types of plots and charts.
https://www.scilab.org/scilab/

TEXT BOOKS / REFERENCES:

TEXT BOOKS :

4. Sant Sharan Mishra, "Computer Oriented Numerical and Statistical Methods", PHI Learning Private Limited, 2013.
5. Rizwan Butt, "Introduction to Numerical Analysis Using Matlab", Infinity Science Press LLC, 2008
6. Ralph P Grimaldi, B.V.Ramana, "Discrete & Combinatorial Mathematics, An Applied Introduction" 5th Edition, Pearson Education, 2009.

REFERENCES:

5. D.S. Chandrasekharaiah, Discrete Mathematical Structures, 4th Edition, PRISM Pvt. Ltd. 2012.
6. Bondy and U.S.R.Murty: Graph Theory and Applications (Freely downloadable from Bondy's website; Google-Bondy)
7. S. Kumarsean, "Linear Algebra A geometric approach", Prentice Hall of India Private Limited, 2001.
8. Kenneth H Rosen, "Discrete Mathematics & its Applications" 7th edition, McGraw-Hill, 2010.

ADDITIONAL LEARNING SOURCES:

7. http://www.personal.kent.edu/~rmuhamma/GraphTheory/graphTheory.htm
8. http://www.tutorialspoint.com/discrete_mathematics/
9. http://www.maths.lu.se/english/library/e-resources/
10. http://sunzi.lib.hku.hk/ER/detail/hkul/3743848
11. https://www.**math**.ucdavis.edu/~**linear**/**linear**-guest.pdf

CO-PO MAPPING:

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

II Semester

Department of Master of Computer Applications[MCA]							
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	Advanced Java Programming	Java Programming

COURSE ASSESSMENT METHOD:

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

COURSE OUTCOMES:

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

CO1: Apply the concepts of J2EE and JDBC in enterprise application development.
CO2: Write Server side applications using Servlets by demonstrating different concepts of Servlets.
CO3: Design and develop web applications using JSP.
CO4: Design and develop components using Java Beans and apply Enterprise Java Bean (EJB) concepts
CO5: Demonstrate and apply the concept of Spring framework for application development.

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 – Packages, JAR files, Java Beans and Enterprise Java Beans (EJB) 12 Hours

Creating packages, JAR files. 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.
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.

UNIT 5 – Introduction to Spring Framework 12 Hours

Problems with the Traditional Approach to J2EE, Lightweight Frameworks, Enter Spring, Spring's values, Spring in Context: Technologies- Inversion of Control and Dependency Injection, Aspect-Oriented Programming, Consistent Abstraction, Resource Management, Techniques. Relationship to Other Frameworks: Persistence Frameworks, Web Frameworks, AOP Frameworks, Other Frameworks. Architecting Applications with Spring: The Big Picture, Persistence and Integration-Database, Data Access Objects. Persistent Domain Objects, Persistence Technology, Business Service Objects, Presentation, Web Tier, Remote Access Facade. A simple program in Spring framework.

TEXT BOOKS / REFERENCES:

Text Books:

2. Marty Hall, Larry Brown. Core Servlets and Java Server Pages. Volume 1: Core Technologies. Second Edition. 2014.
3. Java 6 Programming Black Book, Dreamtech Press. 2012.
4. Andrew Lee Rubinger, Bill Burke. Developing Enterprise Java Components. Enterprise JavaBeans 3.1.O'reilly. 2010.
5. Rod Johnson, Juergen Hoeller, Alef Arendsen, Thomas Risberg, Colin Sampaleanu. Professional Java Development with the Spring Framework, John Wiley, Wrox publications, 2005.

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, Ninth Edition. Comprehensive coverage of the Java Language. Tata McGraw-Hill Edition – 2014.

3. Craig Walls. Spring in Action.4th Edition, Manning-Dreamtech Press-2015

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. http://w3schools.invisionzone.com/index.php?showforum=52
6. http://www.javagenious.com/search/jsp-tutorial-w3schools/
7. *http://gsraj.tripod.com/java/java_tutorial.html*
8. *http://docs.spring.io/spring/docs/current/spring-framework-reference/html/overview.html*
9. *http://courses.springframework.guru/p/introduction-to-spring*
10. *http://spring.org*

CO-PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	H	H	H	H	H	H	M	M	M	H	M
CO2	H	M	H	M	M	M	H	L	M	M	M	L
CO3	H	M	H	M	M	M	H	L	M	M	M	L
CO4	H	M	M	M	M	M	H	L	M	M	M	L
CO5	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	
I	II	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA 220	Analysis and Design of Algorithms using C++	--

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: Apply object oriented techniques to solve bigger computing problems

CO 2: Explore the knowledge of computational complexity, approximation and randomized algorithms

CO 3: Analyze the range of the algorithm and the notion of tractable and intractable problems
CO 4: Design and analyze a wide range of searching and sorting algorithms
CO 5: Implementation of graph and matching algorithms

TOPICS COVERED:

UNIT 1 - 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 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 3 - 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 4 - 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 5 - 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.

Text Book:

1. Anany Levitin: Introduction to the Design and Analysis of Algorithms, Pearson Education, 2003.
2. Herbert Scheldt: The Complete Reference C++, 6th Edition, Tata McGraw Hill 2013.

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.

ADDITIONAL LEARNING SOURCES:

1. https://www.tutorialspoint.com/data_structures_algorithms/
2. http://nptel.ac.in/courses/106101060/

CO-PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	H	H	H	H	H	H	M	M	M	H	M

	CO2	H	M	H	M	M	M	H	L	M	M	M	L
	CO3	H	M	H	M	M	M	H	L	M	M	M	L
	CO4	H	M	M	M	M	M	H	L	M	M	M	L
	CO5	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	
I	II	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA230	Software Engineering	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:

- **CO1:** Elicit, analyze and specify software requirements through a productive working relationship with project stakeholders.
- **CO2:** Make judgment about applicability of techniques in some project.
- **CO3:** Use agile and techniques in application development.
- **CO4:** Apply software testing and quality assurance techniques at the module level, and incorporate these techniques at the system and organization level.
- **CO5:** Apply agile concepts to different case studies

TOPICS COVERED:

UNIT 1 - Introduction to Agile software Development: 12 Hours

Overview, Objectives ,Three Perspectives on Software Engineering , The Agile Manifesto , Individuals and Interactions over Processes and Tools, Working Software over Comprehensive Documentation, Customer Collaboration over Contract, Negotiation, Responding to Change over Following a Plan,Application of Agile Software Development , Data About Agile Software Development,Agile Software Development in Learning Environments University Course Structure, Teaching and Learning Principles, The Studio Environment, The Academic Coach Role ,Overview of the Studio Meetings. Teamwork:Overview, Objectives, A Role Scheme in Agile Teams,Remarks on the Implementation of the Role Scheme, Human Perspective on the Role Scheme, Using the Role Scheme to Scale Agile Projects, Dilemmas in Teamwork, Teamwork in Learning Environments, Teaching and Learning Principles,Role Activities, Student Evaluation.Customers and Users:Overview, Objectives, The Customer, Customer Role, Customer Collaboration, The User, Combining UCD with Agile Development, Customers and Users in Learning Environments, Teaching and Learning Principles, Customer Stories.

UNIT 2 - Time **10 Hours**

Overview, Objectives, Time-Related Problems in Software Projects, List of Time-Related Problems of Software Projects. the Time Perspective, Tightness of Software Development Methods, Sustainable Pace, Time Management of Agile Projects, Time Measurements, Prioritizing Development Tasks, Time in Learning Environments, The Planning Activity, Teaching and Learning Principles, Students' Reflections on Time-Related Issues, The Academic Coach's Perspective. Measures: Overview, Objectives, Why Are Measures Needed, Who Decides What Is Measured? What Should Be Measured, When Are Measures Taken? How Are Measures Taken? Who Takes the Measures? How Are Measures Used? Case Study, Monitoring a Large-Scale Project by Measures, Measure Definition, Measure Illustration, Measures in Learning Environments, Teaching and Learning Principles, Measurement Activities.

UNIT 3–Quality **10 Hours**

Overview, Objectives, The Agile Approach to Quality Assurance, Process Quality, Product Quality, Test-Driven Development, How Does TDD Help to Overcome Some of the Problems Inherent in Testing, Learning: Overview, Objectives, Study Questions, How Does Agile Software Development Support Learning Processes, Agile Software Development from the Constructivist Perspective, The Role of Short Releases and Iterations in Learning Processes, Learning in Learning Environments, Gradual Learning Process of Agile Software Engineering, Learning and Teaching Principle, The Studio Meeting, End of the First Iteration, Intermediate Course Review and Reflection, Abstraction: Overview, Objectives, Study Questions, Abstraction Levels in Agile Software Development, Roles in Agile Teams.

UNIT 4 - Planning: **10 Hours**

The Stand-Up Meeting, Design and Refactoring, Abstraction in Learning Environments, Teaching and Learning Principles.Trust:Overview, Objectives,Software Intangibility and Process Transparency, Game Theory Perspective in Software Development, Ethics in Agile Teams, Diversity, Trust in Learning Environments,Teaching and Learning Principle.Globalization:Overview, Objectives, Study Questions, The Agile Approach in Global Software Development, Communication in Distributed Agile Teams, Planning in Distributed Agile Projects, Case Study, Tracking Agile Distributed Projects, Reflective Processes in Agile Distributed Teams, Organizational Culture and Agile Distributed Teams, Application of Agile Principles in Non-Software Projects.

UNIT 5–Reflection **10 Hours**

Overview, Objectives, Case Study, Reflection on Learning in Agile Software Development, Reflective Practitioner Perspective, Retrospective, The Retrospective Facilitator, Case Study, Guidelines for a Retrospective Session, Application of Agile Practices in Retrospective Sessions, End of the Release Retrospective, Reflection in Learning Environments. Change: Overview, Objectives, A Conceptual Framework for Change Introduction, Changes in Software Requirements, Organizational Changes, Transition to an Agile Software Development Environment. Leadership: Overview, Objectives, Leaders, Leadership Styles, Case Study, The Agile Change Leader, Coaches, Leadership in Learning Environments, Teaching and Learning Principles. Delivery and Cyclicality: Overview, Objectives,

Delivery, Towards the End of the Release, Release Celebration, Reflective Session Between Releases, Cyclicality ,Delivery and Cyclicality in Learning Environments, The Delivery in the Studio, Teaching and Learning Principles.

TEXT BOOKS / REFERENCES:

Text books:

Orit Hazzan and Yael Dubinsky, Agile Software Engineering, Springer, 2009

Reference books:

1. Cockburn, Agile Software Development, Pearson Education India
2. Mike Cohn, Agile Estimating and Planning, Pearson Education, 2005
3. Michele Sliger, StaciaBroderick, The Software Project Manager's Bridge to Agility, Addison-Wesley Professional, 2008

ADDITIONAL LEARNING SOURCES:

*1.*www.allaboutagile.com/what-is-agile-10-key-principles/

2.https://www.versionone.com/agile-101

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

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	
I	II	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA240	Operating System with Linux	--

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, the student will be able to:

CO1: Recognize the structure of operating system, interaction of an operating system and application programs.

CO2: Analyze the various programming paradigms viz., multi-process and multi-threaded programming.

CO3: Examine the various resource and memory management techniques.

CO4: Distinguish the different features of real time and mobile operating systems.

CO5: Identify current issues in system security; demonstrate various factors can influence the overall performance of an operating system.

TOPICS COVERED:

UNIT 1- Computer and Operating Systems Structure 11 Hours

Basic Elements, Processor Registers, Instruction Execution, The Memory Hierarchy, Cache Memory, I/O Communication Techniques, Introduction to Operating System, Mainframe Systems, Desktop Systems, Multiprocessor Systems, Distributed Systems, Clustered Systems, Real - Time Systems, Handheld Systems, Feature Migration, Computing Environments.
System Structures: System Components, Operating – System Services, System Calls, System Programs, System Structure, Virtual Machines, System Design and Implementation, System Generation

UNIT 2 - Process Management and Mutual Execution 10 Hours

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

UNIT 3 - Deadlock and Memory Management 11 Hours

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

UNIT 4 - File System and Secondary Storage 10 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**,** Disk Structure, Disk Scheduling, Disk Management.

UNIT 5 - Computer Security and Case study of Linux Operating system 10 Hours

The Security Problem, User Authentication, Program Threats, System Threats. Linux System Linux history, Design Principles, Kernel modules, Process, management, scheduling, Memory management, File systems, Input and output, Inter-process communications.

TEXT BOOKS/REFERENCES:

TEXT BOOKS :

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating System Principles, 8th edition, Wiley India, 2009.
5. William Stallings, "Operating System Internals and Design Principles" Pearson, 6thedition, 2012

REFERENCES:

1. Dhananjay M. Dhamdhere, "Operating Systems – A Concept – Based Approach", Tata McGraw – Hill, 3rd Edition, 2012.
2. Harvey M Deital: Operating systems, 3rd Edition, Pearson Education, 1990.
3. Chakraborty , "Operating Systems" Jaico Publishing House, 2011

ADDITIONAL RESOURCES:

1. https://www.tutorialspoint.com/operating_system/os_linux
2. https://www.cs.uic.edu/~jbell/CourseNotes/OperatingSystems.

CO-PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	L	M	L	M	M	M	M	M	M
CO2	H	H	M	H	M	M	M	H	M	M	M	H
CO3	H	M	M	H	M	M	H	H	H	H	H	M
CO4	M	H	L	M	M	H	H	M	M	M	H	H
CO5	M	M	H	M	H	H	M	M	H	H	M	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	
I	II	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA250	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 able to:

CO 1: Master the basic concepts of data models, modeling notations, SQL and construct queries in SQL.

CO 2: Explore the relation with Algebraic expressions, operations, transactions in Relational Database Systems.

CO 3: Explore the concepts of transactions and transaction processing in data base applications

CO 4: Design the queries to handle all the basic and advanced operations of RDBMS on an application

CO 5: Design and build a simple database system and demonstrate the competence with the fundamental tasks involved with Modeling, Design and Implementing a RDMS

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:

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

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**	
I	II	Laboratory	01	00	03	00	48
Course No		**Course Title**				**Pre Requisites**	

MCA21L	Advanced Java Programming Laboratory	Java Programming

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, students will be able to:

CO1: Apply the concepts of J2EE and JDBC in enterprise application development.
CO2: Write Server side applications using Servlets by demonstrating different concepts of Servlets.
CO3: Design and develop web applications using JSP.
CO4: Design and develop components using Java Beans and apply Enterprise Java Bean (EJB) concepts
CO5: Demonstrate and apply the concept of Spring framework for application development.

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, UPDATE, DELETE records to the Database using Statement Object
4. Write a JDBC program to INSERT, UPDATE, DELETE records to the Database using PrepareStatement Object
5. Write a JDBC program to INSERT, UPDATE, DELETE records to the Database using Statement Object With Batch Insert, Update & Delete.
6. Write a JDBC program to INSERT, UPDATE, DELETE records to the Database using PrepareStatement Object with Batch Insert, Update & Delete.
7. Write a servlet program to display hello world.
8. Write a servlet program to generate a plain text and HTML output.
9. Write a servlet program to generate lottery numbers
10. Write a servlet program to demonstrate single thread model.
11. Write a servlet program to read three parameters from Form Data
12. Write a servlet program to read all data from Form data.
13. Write a servlet program to demonstrate response header.
14. Write a servlet program to show request header.
15. Write a servlet program to display number of visitors visited the page.
16. Write a servlet program to handle cookies.
17. Write a servlet program to demonstrate session tracking
18. Create a simple Login application using HTML, MySQL, Servlet and Web-XML.
19. Create a simple User Registration application using HTML, MySQL, Servlet and Web-XML.
20. Write a Servlet program to generate an output in different MIMEs (Excel, HTML, PDF etc.)
21. Write a JSP program to read three parameters from another JSP page and display.

22. Write a JSP program to display Sales entry information.
23. Write a JSP program to test the background color of the page
24. Write a JSP program to generate access counter.
25. Write a JSP program which conditionally generates a spreadsheet
26. Write a program to generate a Plug in Applet using JSP
27. Write a program to generate a string bean using JSP
28. Write a program to create a Excel sheet using JSP
29. Write a program to create a Bean sheet using
30. Write a program to create a reusable footer using JSP
31. Write a program to count the shared counts using JSP.
32. Write a Program which includes files using JSP
33. Write a JSP program which generates the wishes for the day based on time.
34. Write a JSP program to generate random numbers.
35. Write a JSP program to generate a order confirmation by taking the order details from the user.
36. Write a JSP program which should compute the speed.
37. 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.
38. 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)
39. Design and develop any application of your own using EJB Components (Session, Message Driven and Entity Bean)
40. Design and develop a simple application using spring frame work.

TEXT BOOKS / REFERENCES:

Text Books:

1. Marty Hall, Larry Brown. Core Servlets and Java Server Pages. Volume 1: Core Technologies. Second Edition.
2. Java 6 Programming Black Book, Dreamtech Press. 2012.
3. Andrew Lee Rubinger, Bill Burke. Developing Enterprise Java Components. Enterprise JavaBeans 3.1.O'reilly.
4. Rod Johnson, Juergen Hoeller, Alef Arendsen, Thomas Risberg, Colin Sampaleanu. Professional Java Development with the Spring Framework, John Wiley, Wrox publications, 2005.

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. Craig Walls. Spring in Action.4th Edition, Manning-Dreamtech Press-2015

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:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	H	H	H	H	M	M	M	H	M
CO2	H	M	H	M	H	M	H	L	M	M	M	L
CO3	H	M	H	M	H	M	H	L	M	M	M	L
CO4	H	M	M	M	H	M	H	L	M	M	M	L
CO5	H	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	
I	II	Laboratory	01	00	03	00	48

Course No	Course Title	Prerequisites
MCA22L	Algorithms with C++ Laboratory	C Programming

COURSE ASSESSMENT METHOD:

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

COURSE OUTCOMES:

CO 1: Apply object oriented techniques to solve bigger computing problems
CO 2: Explore the knowledge of computational complexity, approximation and randomized algorithms
CO 3: Analyze the range of the algorithm and the notion of tractable and intractable problems
CO 4: Design and analyze a wide range of searching and sorting algorithms
CO 5: Implementation of graph and matching algorithms

LIST OF EXPERIMENTS COVERED:

1. Implement Recursive Binary search and Linear search and determine the time required to search an element. Repeat the experiment for different values of n, the number of elements in the list to be searched and plot a graph of the time taken versus n.

2. Sort a given set of elements using the Heapsort method and determine the time required to sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n.

3. Sort a given set of elements using Merge sort method and determine the time required to sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n.

4. Sort a given set of elements using Selection sort and determine the time required to sort elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n.

5. Implement 0/1 Knapsack problem using dynamic programming.

6. From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm.

7. Sort a given set of elements using Quick sort method and determine the time required sort the elements. Repeat the experiment for different values of n, the number of elements in the list to be sorted and plot a graph of the time taken versus n.

8. Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal's algorithm.

9. a. Print all the nodes reachable from a given starting node in a digraph using BFS method.
   b. Check whether a given graph is connected or not using DFS method.

10. Find a subset of a given set S = {sl,s2,.....,sn} of n positive integers whose sum is equal to a given positive integer d. For example, if S= {1,2, 5, 6, 8} and d = 9 there are two solutions{1,2,6}and{1,8}.A suitable message is to be displayed if the given problem instance doesn't have a solution.

11. a. Implement Horspool algorithm for String Matching.
    b. Find the Binomial Co-efficient using Dynamic Programming.

11. Find Minimum Cost Spanning Tree of a given undirected graph using Prim's algorithm.

13. a. Implement Floyd's algorithm for the All-Pairs- Shortest-Paths problem.
    b. Compute the transitive closure of a given directed graph using Warshall's algorithm.

14. Implement N Queen's problem using Back Tracking.

TEXT BOOKS/ REFERENCES :

Text Books:

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

Reference Books:

1. Data structures, Algorithms and Applications in C++, S. Sahani, University Press (India) Pvt.Ltd, 2nd Edition.

ADDITIONAL LEARNING SOURCES:

1. https://www.tutorialspoint.com/data_structures_algorithms/
2. http://nptel.ac.in/courses/106101060/

CO-PO MAPPING:

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

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	
I	II	Laboratory	01	00	03	00	48

Course No	Course Title	Pre Requisites
MCA23L	Software Engineering Laboratory	Management Information System

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: Elicit, analyze and specify software requirements through a productive working relationship with project stakeholders.
CO2: Make judgment about applicability of techniques in some project.
CO3: Use agile and techniques in application development.
CO4: Apply software testing and quality assurance techniques at the module level, and incorporate these techniques at the system and organization level.
CO5: Apply agile concepts to different case studies

LIST OF EXPERIMENTS COVERED:

Case Studies

1. Case Studies of Metaphor Use
2. Software Organizational Survey from
3. An Iteration Timetable of an Agile Team
4. Role-Related Measures.
5. Reflection on TDD , Measured TDD
6. Quality in Learning Environments
7. Size and Complexity Measures

8. Abstraction During Iteration
9. Book Writing, Globalization in Learning Environments, Teaching and Learning Principles,
10. Follow-the-Sun with Agile Development.
11. A Report of an Organizational Survey
12. Applying an Agile Process to a Transition Process

Based on case studies develop application.

Project Work

Objective: Understand Importance of SDLC approach &various processes.
Problem Description: Software Development Life cycle involve multiple phases like Requirement analysis, Design, Coding, Testing & Maintenance.In each phase certain activities are performed to deliver the deliverables or outcome eg. From Requirement Analysis phase outcome is SRS document which is input to the Design phase.Project group is assigned with a project and asked to perform the complete cycle of SDLC.

Following are some instruction related to the project work,
1. Project group has to collect the required information related to the assigned system/project.
2. Project group has to develop the project and deliver the deliverables on the timelines given by the Teacher.
3. On the completion of duration, each project group has to submit a project report. The teacher will evaluate the report.
4. The Implementation of project will be done Pseudo code
5. Report submission.

Work Tasks	Assigned Persons	Remarks
Requirement Gathering		
Requirement Analysis		
Object Oriented analysis		
Coding		
Testing		
Reviews and Check Lists		
Project Completed		

TEXT BOOKS / REFERENCES:

Text books:

Orit Hazzan and Yael Dubinsky,Agile Software Engineering,Springer 2009

Reference books:

1. Cockburn, Agile Software Development, Pearson Education India
2. Mike Cohn, Agile Estimating and Planning, Pearson Education, 2005
3. Michele Sliger, Stacia Broderick,The Software Project Manager's Bridge to Agility, Addison-Wesley Professional, 2008

ADDITIONAL LEARNING SOURCES:

*1.*www.allaboutagile.com/what-is-agile-10-key-principles/
2.https://www.versionone.com/agile-101

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

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	
I	II	Laboratory	01	00	03	00	48

Course No	Course Title	Pre Requisites
MCA24L	Operating System and Linux Laboratory	---

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:

CO1: Analyze and use basic commands and functionalities of the operating system.
CO2: Use various operating system calls and synchronization libraries in software/ hardware interfaces.
CO3: Examine use of threads over process and implement synchronized programs using multithreading concepts.
CO4: Analyze and simulate CPU scheduling algorithms like FCFS, Round Robin, SJF, and Priority.
CO5: Recognize memory management schemes and simulate page replacement schemes.

LIST OF EXPERIMENTS COVERED:

1. Execution of various file/directory handling commands.
2. Simple shell script for basic arithmetic and logical calculations.
3. Shell scripts to check various attributes of files and directories.
4. Shell scripts to perform various operations on given strings.
5. Shell scripts to explore system variables such as PATH, HOME etc.
6. Shell scripts to check and list attributes of processes.
7. Execution of various system administrative commands.

8. Write awk script that uses all of its features.
9. Use seed instruction to process /etc/password file.
10. Write a shell script to display list of users currently logged in.
11. Write a shell script to delete all the temporary files.
12. Write a shell script to search an element from an array using binary searching.
13. Write a C program to simulate the following non-preemptive CPU scheduling algorithms to find turnaround time and waiting time.
    a) FCFS b) SJF c) Round Robin d) Priority
14. Write a C program to simulate multi-level queue scheduling algorithm considering the following scenario. All the processes in the system are divided into two categories – system processes and user processes. System processes are to be given higher priority than user processes. Use FCFS scheduling for the processes in each queue.
15. Write a C program to simulate the following file allocation strategies.
    a) Sequential b) Indexed c) Linked
16. Write a C program to simulate the MVT and MFT memory management techniques.
17. Write a C program to simulate the following contiguous memory allocation techniques
    a) Worst-fit b) Best-fit c) First-fit
18. Write a C program to simulate paging technique of memory management.
19. Write a C program to simulate the following file organization techniques
    a) Single level directory b) Two level directory c) Hierarchical
20. Write a C program to simulate Bankers algorithm for the purpose of deadlock avoidance
21. Write a C program to simulate disk scheduling algorithms
    a) FCFS b) SCAN c) C-SCAN
22. Write a C program to simulate page replacement algorithms
    a) FIFO b) LRU c) LFU
23. Write a C program to simulate page replacement algorithms
    a) Optimal
24. Write a C program to simulate producer-consumer problem using semaphores
25. Write a C program to simulate the concept of Dining-Philosophers problem.

TEXT BOOKS/REFERENCES:

TEXT BOOKS:

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating System Principles, 8th edition, Wiley India, 2009.
2. William Stallings, "Operating System Internals and Design Principles" Pearson, 6thedition.

REFERENCES:

1. Dhananjay M. Dhamdhere, "Operating Systems – A Concept – Based Approach", Tata McGraw – Hill, 3rd Edition, 2012.
2. Harvey M Deital: Operating systems, 3rd Edition, Pearson Education, 1990.
3. Chakraborty , "Operating Systems" Jaico Publishing House, 2011

ADDITIONAL RESOURCES:

1. https://www.tutorialspoint.com/operating_system/
2. http://www.scoopworld.in/search/label/CSE

CO-PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	M	M	M	M	M	H	L	L	H
CO2	H	M	M	H	M	L	M	H	H	H	H	M
CO3	M	M	M	H	M	L	H	H	M	H	M	M
CO4	M	M	M	H	H	M	H	M	M	M	M	M
CO5	M	M	M	H	H	H	L	L	L	M	H	H
CO6	H	H	M	H	M	H	M	L	L	H	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	
I	II	Laboratory	01	00	03	00	48

Course No	Course Title	Pre Requisites
MCA25L	Relational Database Management System Laboratory	Data Structures

COURSE ASSESSMENT METHOD:

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

COURSE OUTCOMES:

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

CO 1: Master the basic concepts of data models, modeling notations, SQL and construct queries in SQL.

CO 2: Explore the relation with Algebraic expressions, operations, transactions in Relational Database Systems.

CO 3: Explore the concepts of transactions and transaction processing in data base applications

CO 4: Design the queries to handle all the basic and advanced operations of RDBMS on an application

CO 5: Design and build a simple database system and demonstrate the competence with the fundamental tasks involved with Modeling, Design and Implementing a RDMS

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: int, from: string, to: string, distance: int, 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:

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

III Semester

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**	
II	III	Theory	04	04	00	00	52
Course No		**Course Title**				**Pre Requisites**	

MCA310	Web programming	Programming 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,

CO1: Apply basic design and usability principals when creating content for the internet
CO2: Demonstrate practical skills in workflows and methods used to create websites within a professional environment.
CO3: Make informed and critical decisions regarding client development using Modules.
CO4: Design and implement reasonably sophisticated server-side applications using one or more suitable technologies.
CO5: Have the skills to critically analyze and evaluate web applications. Construct websites using a variety of skills and techniques.

TOPICS COVERED:

UNIT 1 -HTML5 **10 Hours**

A web standards approach,Your first Plunge into HTML5,New Structural Elements, Aricher approach to Marking Up Content, Rich Media

UNIT 2 – Introduction **12 Hours**

Java script you need to know, JavaScript Primer, Including Scripts on a Page, Statements, Functions, Parameters and Return Values, Types and Variables, Primitive Types, Strings, numbers, undefined null, javascript operators, working with objects, enumerating properties, control flow, conditional statements, working with arrays, JSON ,The basics Angular JS,Framework, Introduction to MVC,Design patterns,model view controller.

UNIT 3 – Filters and Modules **10 Hours**

Introduction to filters,Built in filters, Angular JS modules,Directives,the basic of directives,angular socks,Event handling directives,creating custom directive,the restrict option,the template option,the link option,Working with forms,HTML Forms Overview,Angular JS Forms,validating forms.

UNIT 4–Services and Server Communication **10 Hours**

Using Services, Server Communication, Handling Returned Data, Organizing Views, Using URL Routes, AngularJS Animation, Applying Animations, Deployment Considerations, Configuration, Testing, Error Handling**,** Working with forms ,services and server communication,Organizing Views,Angular animation,Deployment considerations.

UNIT 5– PHP **10 Hours**

Getting up and running up with PHP, introducing PHP,first PHP script,Learning the language,language basics,decisions and loops,strings, Arrays, Functions, objects.

TEXT BOOKS / REFERENCES:

Text books:

1. Christopher Murphy, Richardclark, OliStudholme, Divys Manian, Beginning HTML5 and CSS3, Apress Publication,2012
2. **Grant**, Andrew ,Beginning AngularJS, Apress Publication,2014
3. Matt Doyle, Beginning PHP 5.3,Wiley Publisher,2010

Reference books:

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

ADDITIONAL LEARNING SOURCES:

1. www.w3schols.com
2. www.afterhoursprograming.com

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

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	
II	III	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA 320	Intelligent Algorithms	ADA

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: Analyze a variety of algorithms with practical applications.
CO2: Apply various techniques to solve new problems that may arise in various domains.
CO3: Analyze various string matching, randomized & graph algorithms and explore their applications.
CO4: Explore the basic elements of intelligent web applications.
CO5: Implement and evaluate the recommendation engines.

TOPICS COVERED:

UNIT: 1 - Review of Analysis Techniques: 12 Hours
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: 10 Hours
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 and Randomized Algorithms: 10 Hours
Naïve string Matching; Rabin - Karp algorithm; String matching with finite automata; Knuth-Morris-Pratt algorithm; Boyer – Moore algorithms. Randomizing deterministic algorithms, Monte Carlo and Las Vegas algorithms.

UNIT: 4 - Intelligent Web: 10 Hours
Examples of intelligent web applications; Basic elements of intelligent applications; What applications can benefit from intelligence?; How can I build intelligence in my own application?; Machine learning, data mining, and all that. Eight fallacies of intelligent applications. Searching : Searching with Lucene; Why search beyond indexing?; Improving search results based on link analysis; Improving search results based on user clicks; Ranking Word, PDF, and other documents without links; Large-scale implementation issues; Is what you got what you want? Precision and recall.

UNIT:5 - Creating Suggestions and Recommendations: 10 Hours
An online music store: the basic concepts; How do recommendation engines work?; Recommending friends, articles, and news stories; Recommending movies on a site such as Netflix.com; Large-scale implementation and evaluation issues.

Text Book/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.
3. Haralambos Marmanis and Dmitry Babenko, Algorithms of the Intelligent Web, Manning Publications, 2009.

Reference Books:

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

CO-PO MAPPING:

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

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	
II	III	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA330	Object-Oriented Modeling and Design	Software Engineering

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: Transform Use Cases into Object Oriented software Realizations through OO Analysis and OO Design.
CO2: Document your requirements, analysis, and design models in the Unified Modeling Language (UML) notation.
CO3: Apply techniques of state machines and design patterns to the system.
CO4: Specify, analyze and design the use case driven requirements for a particular system.
CO5: Identify and analyze the various subsystems, components and collaborate them interchangeably.

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. SystemConception: 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. Patterns What is a pattern and what makes a pattern? Pattern categories; Relationships between patterns; Pattern description. Design PatternsIntroduction, structural decomposition, Organization of work, Model View Controller; Communication Patterns: Forwarder-Receiver; Client-Dispatcher-Server; Publisher-Subscriber; Management Patterns: Command processor; Whole Part, Master Slave, View Handler;

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. Frank Buschmann, RegineMeunier, Hans Rohnert, Peter Sommerlad, Michael Stal: Pattern-Oriented Software Architecture, A System of Patterns, Volume 1, John Wiley and Sons, 2006. (Chapters 1, 3)

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:

	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

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	
II	III	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA340	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 able to:

CO1: Analyze and distinguish the basic concepts, principles and techniques of data communication along with the layers of OSI and TCP/IP model.

CO2: Independently understand and distinguish the concept of links, nodes and data transmission issues in the network.
CO3: Capability to categorize wired LANs: Ethernet, IPv4 addresses and performance of The network-layer.
CO4: Design and demonstrate the services of TCP and UDP.
CO5: Ability to summarize and interpret the basic concepts of Application-Layer paradigms and standard client-server protocols.

TOPICS COVERED:

UNIT 1 - Basics of Data Communications and Physical Layer 10 Hours

Data Communications: Components, Data Representation, Data Flow, Networks; Network Criteria, Physical Structures, Network Types: LAN, WAN, Switching, 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, Introduction to Physical Layer, Data and Signals, Periodic Analog Signals, Digital Signals, Transmission Impairment, Data Rate Limits, Performance, Switching: Circuit-Switched Networks, Packet Switching, Datagram Networks, Virtual Circuit Networks.

UNIT 2 - Data Link Layer 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 3 - LANs and Network Layer 10 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, 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.

UNIT 4 - 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 5 - Application Layer and Standard Client-Server Protocols 12 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, 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.

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. 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.

ADDITIONAL LEARNING SOURCES:

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

CO-PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	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

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
II	III	Theory (Stream)	05	03	00	02	65

Course No	Course Title	Pre - Requisites
MCA351	Computer Vision and Pattern Recognition	--

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, the student will be able to:

CO1: Apply knowledge and Understanding of computer vision and pattern recognition considering applications in various fields.
CO2: Making judgments in choosing and incorporating the concept of Probability theory, Baye's theorem in PR applications.
CO3: Autonoumsly extend the knowledge in designing how patterns are represented, feature are extracted appropriately and extending the same to data reduction algorithms.
CO4: Develop the algorithms to design the hybrid classifiers by understand the working of SVMs, Neural Networks Architectures.
CO5: Design of Computer Vision model for a real time application.

TOPICS COVERED:

UNIT-1 Introduction to Computer Vision, Image Formation, PR & Probability 12Hours

A brief history of computer vision; Geometric primitives and transformations; Photometric image formation; The digital camera; Machine perception, an example; Pattern Recognition System; The Design Cycle; Learning and Adaptation; Probabilities of Events; Random Variables; Joint Distributions and Densities; Moments of random variables.

UNIT-2 Representation & Nearest Neighbour Based Classifiers 10 Hours

Data Structures for Pattern Representation; Representation of Clusters; Proximity Measures; Size of Patterns Abstractions of the Data Set; Feature Extraction; Feature Selection; Evaluation of Classifiers; Evaluation of Clustering. Nearest Neighbour Algorithm;Variants of the NN Algorithm; Use of the Nearest Neighbour Algorithm for Transaction Databases; Efficient Algorithms ; Data Reduction; Prototype Selection.

UNIT-3 Bayes Classifier & Decision Trees 10 Hours

Bayes Theorem; Minimum Error Rate Classifier; Estimation of Probabilities; Comparison with the NNC; Naive Bayes Classifier; Bayesian Belief Network; Introduction to Decision Tress; Decision Trees for Pattern Classification; Construction of Decision Trees; Splitting at the Nodes; Overfitting and Pruning; Example of Decision Tree Induction.

UNIT-4 Support Vector Machines & Combination of Classifiers **10 Hours**

Introduction to SVM; Learning the Linear Discriminant Function; Neural Networks; SVM for Classification; Methods for Constructing Ensembles of Classifiers; Methods for Combining Classifiers

UNIT-5 Clustering & An Application: Handwritten Digit Recognition **10 Hours**

Importance of Clustering; Hierarchical Algorithms; Partitional Clustering; Clustering Large Data Sets; Description of the Digit Data; Pre-processing of Data; Classification Algorithms; Selection of Representative Patterns; Results.

TUTORIALS:

1. Simple feature extraction techniques to understand pattern recognition system.
2. Concept of Probability theory, Baye's theorm & Bayesian belief networks.
3. Nearest Neighbour Algorithm, Variants of NN, Case study to Transaction Database.
4. Naïve Bayes Classifier, Bayesian Belief Network, Pattern Classifiers based on decision trees.
5. Implementation of SVM and Neural Network Architectures.
6. Design and development of clustering approaches and recognition system for handwritten characters.

TEXT BOOKS/ REFERENCES:

Text Books:

1. Richard Szeliski, Computer Vision: Algorithms and Applications, Springer, 2010.
2. M.N. Murthy and V. Susheela Devi, Pattern Recognition: An Algorithmic Approach, Springer, 2011.

References:

1. Richard O. Duda, Peter E. Hart, and David G.Stork: Pattern Classification, 2nd Edition, Wiley-Interscience, 2001.
2. Earl Gose, R. Johnsonbaugh and S. Jost, Pattern Recognition and Image Analysis, PHI, 2009.

ADDITIONAL LEARNING SOURCES:

1. **www.journals.elsevier.com/pattern-recognition/**

CO-PO MAPPING:

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

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
II	III	Theory (Stream)	05	03	00	02	65

Course No	Course Title	Pre Requisites
MCA352	Data Warehouse	Relational Database Management 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:

CO1: Design a data warehouse or data mart to present information needed by management in a form that is usable for management client & Comprehend several data preprocessing methods.

CO2: Ability to do Conceptual, Logical, and Physical design of Data Warehouse

CO3: Able to produce and document dimensional models for a data warehouse based on an informal domain description.

CO4: Utilize the concept of data warehouse and OLAP for data Warehousing and tools.

CO5: xtrapolate knowledge and skills to design a data warehouse to support and provide business solutions

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. The need for data warehousing, paradigm shift, business problem definition, operational and informational data stores, characteristics.Overview of client/server architecture, server specialization in client/server computing environments, server functions, server hardware architecture, system considerations, risc versus cisc, multiprocessor systems.

Unit 2 - Data Warehousing Components **10 Hours**

Overall Architecture, data warehouse database, sourcing, acquisition, cleanup and transformation tools, metadata, access tools, data marts, data warehouse administration and management.Business Considerations, design considerations, technical considerations, implementation considerations, benefits of data warehousing.

Unit 3 - Mapping the data warehouse **10 Hours**

Relational database technology for data warehouse, types, database architectures for parallel processing, parallel RDBMS features, alternative technologies, parallel dbms vendors, data layouts for best access, multidimensional data models, bitmapped indexing, complex data types.

Unit 4 - Data Extraction, Cleanup and Transformation Tools **12 Hours**

Tool requirements, vendor approaches, access to legacy data, vendor solutions, transformation engines. Meta data: defined, meta data interchange initiative, metadata repository, metadata management, implementation examples, meta data trends. Need for OLAP, OLAP Guidelines, Categorization of OLAP Tools.

Unit 5 - Business Analysis **08 Hours**

Tool Categories, Need for applications, cognos impromptu, applications, methodology, business intelligence market definition, situation overview, future outlook, essential guidance.

Tutorials:

Case Studies:

1. Data Warehousing Solution for One of Europe's Largest Financial Services Groups.
2. Data Warehousing for a Health Benefits Company
3. Data warehousing solution for banking system.
4. Data warehousing solutions for tax fraud with advanced analytics.
5. Data warehousing solutions for international satellite TV service provider**.**
6. Data warehousing solutions for Correlating data across the business.
7. Business Reporting & Customer Information Datamart Architecture Setup & Roll-out for a global technology company.
8. Global Planning Data Automation.
9. COTS-Anti Money Laundering.
10. Management Information System for Trade Finance.

TEXT BOOKS / REFERENCES:

Text books:

1. Alex Berson, Stephen J smith : Data Warehousing,Data Mining, & OLAP, Tata Mcgraw-Hill, 2012.
2. Gajendra Sharma: Data Mining, Data Warehousing and OLAP, Katson Books, 2010.
3. Jiawei Han and Micheline Kamber: Data Mining - Concepts and Techniques, 2nd Edition, Morgan Kaufmann Publisher, 2012.

Reference books:

1. Paulraj Ponnaiah : Data Warehousing fundamentals for IT professionals, wiley student publishers, second edition , 2014.
2. Ralph Kimball, Margy Ross : The data warehouse toolkit, third edition , wiley publishers, 2012.

ADDITIONAL LEARNING SOURCES:

1. https://www.coursera.org/specializations/data-warehousing.
2. www.knowledge-management-tools.net/data-warehousing.htm.
3. www.slideshare.net/2cdude/data-warehousing.
4. https://www.edx.org/course/delivering-relational-data-warehouse-microsoft-dat216x.

CO-PO MAPPING:

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

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	
II	III	Theory (Stream)	05	03	00	02	65

Course No	Course Title	Pre Requisites
MCA353	Enterprise Resource Planning	MIS, SE

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: Comprehend the basics and concepts of ERP

CO2: Apply different ERP related technologies

CO3: Implement ERP system by utilizing various concepts of ERP

CO4: Analyze different business models of ERP

CO5: Analyze the present and future trends of ERP.

TOPICS COVERED:

UNIT 1 - Introduction **10 Hours**

Enterprise—An Overview, Business Processes, Introduction to ERP, Basics ERP Concepts, Justifying ERP Investments, Risks of ERP, Benefits of ERP.

UNIT 2 - ERP and Technology **10 Hours**

ERP and Related Technologies, Business Intelligence (BI) and Business Analytics (BA), E-Commerce and E-Business, Business Process Reengineering (BPR), Data Warehousing and Data Mining, On-line Analytical Processing (OLAP).

UNIT 3 - ERP and Technology **10 Hours**

Product Life Cycle Management (PLM), Supply Chain Management (SCM), Customer Relationship Management (CRM), Geographic Information System (GIS), Advanced Technology and ERP Security.

UNIT 4 - ERP Implementation **10 Hours**

To be or not to be…, Implementation Challenges, ERP Implementation (Transition) Strategies, ERP Implementation Life Cycle, Pre Implementation Tasks: Getting Ready, Requirements Definition, Implementation Methodologies, ERP Development Methods, Process Definition, Contracts with Vendors, Consultants and Employees, Training and Education, Data Migration, Project Management and Monitoring, Post-Implementation Activities, Success and Failure Factors of an ERP Implementation

UNIT 5 - The Business Modules **12 Hours**

Business Modules of an ERP Packages, Financials, Manufacturing (Production), Human Resource Management, Plant Maintenance, Materials Management, Quality Management, Marketing, Sales, Distribution and Services.

TEXT BOOKS / REFERENCES:

Text books:

1. Alexis Leon, “ERP Demystified”, Tata McGraw Hill, 14 Aug 2014.

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 Fernandz, “The SAP R /3 Hand book”, Tata McGraw Hill, 2006.

TUTORIALS:

Developing Following Applications.

Using any Database Systems.

1. Financial System.
2. Manufacturing System.
3. Human Resource Planning.
4. Plant Maintenance.
5. Materials Management System.
6. Quality Management System.
7. Marketing, Sales, & Distributing System etc..

ADDITIONAL LEARNING SOURCES:

1. http://www.netsuite.com/portal/resource/articles/erp/what-is-erp.shtml
2. https://www.managementstudyguide.com/enterprise-resource-planning-1-articles.htm

CO - PO MAPPING:

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

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	
II	III	Laboratory	01	00	03	00	48

Course No	Course Title	Pre Requisites
MCA31L	Web Programming Laboratory	Programming 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 able to,

CO1: Apply basic design and usability principals when creating content for the internet
CO2: Demonstrate practical skills in workflows and methods used to create websites within a professional environment.
CO3: Make informed and critical decisions regarding client development using Modules.
CO4: Design and implement reasonably sophisticated server-side applications using one or more suitable technologies.
CO5: Have the skills to critically analyze and evaluate web applications. Construct websites using a variety of skills and techniques.

LIST OF EXPERIMENTS COVERED:

HTML

1. Develop a program to create links to other HTML documents or web resources.
2. Develop a program to insert images into the HTML document.
3. Develop a program to create ordered, unordered, definition, nested list.
4. Program on creating an image –map-An image with clickable regions.

AngularJS

1. Program to evaluate expression.
2. Program to evaluate expression using variables.
3. Program to display first name and Last name.
4. Program to evaluate expression with numbers.
5. Program to evaluate expression with objects.
6. Program to evaluate expression with arrays.
7. Program to using a new directive as element.
8. Program to read a static JSON file.
9. Program to display a table with CSS
10. Program to read data from MYSQL database.

PHP

1. Programs on PHP Scripts.
2. Program on decision and loops.
3. Program on strings and arrays.
4. Program on functions.
5. Program on objects.

TEXT BOOKS / REFERENCES:

Text books:

1. Christopher Murphy,Richardclark,OliStudholme,DivysManian,Beginning HTML5 andCSS3,

Apress Publication,2012
2. Grant, Andrew ,Beginning AngularJS, Apress Publication**,**2014
3. Matt Doyle, Beginning PHP 5.3,Wiley Publisher,2010

Reference books:
Chris Bates: Web Programming Building Internet Applications, 3rd Edition, Wiley India, 2007

ADDITIONAL LEARNING SOURCES:

http://php.net/manual/

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

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	
II	III	Laboratory	01	00	03	00	48

Course No	Course Title	Prerequisites
MCA32L	Intelligent Algorithms Laboratory	ADA

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:

CO1: Analyze a variety of algorithms with practical applications.
CO2: Apply such techniques to solve new problems that may arise in various domains.
CO3: Analyze various string matching, randomized & graph algorithms and explore their applications.
CO4: Explore the basic elements of intelligent web applications.
CO5: Understand the implementation and evaluation of recommendation engines.

LIST OF EXPERIMENTS COVERED:

1. Design and develop to implement the Bellman-Ford Algorithm and determine its

performance.

2. Design and develop to implement Johnson's Algorithm
3. Implement Linear modular equation.
4. Implement Monte Carlo algorithm.
5. Implement Naïve string Matching Algorithm.
6. Implement Rabin - Karp algorithm.
7. Develop string matching with finite automata.
8. Design and develop Knuth-Morris-Pratt algorithm and Boyer – Moore algorithms.

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.
3. Haralambos Marmanis and Dmitry Babenko, Algorithms of the Intelligent Web, Manning Publications, 2009.

ADDITIONAL LEARNING SOURCES:

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

CO-PO Mapping:

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

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	
II	III	Laboratory	01	00	03	00	48

Course No	Course Title	Pre Requisites
MCA33L	Object-Oriented Modeling and Design Laboratory	Software Engineering

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

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 using ascertain functions

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.
4. 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

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	
II	III	Laboratory	01	00	03	00	48

Course No	Course Title	Pre Requisites
MCA34L	Computer Networks Laboratory	Computer Networks

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, students will be able to:

CO1: Analyze and distinguish the basic concepts, principles and techniques of data communication along with the layers of OSI and TCP/IP model.
CO2: Independently understand and distinguish the concept of links, nodes and data transmission issues in the network.
CO3: Capability to categorize wired LANs: Ethernet, IPv4 addresses and performance of The network-layer.
CO4: Design and demonstrate the services of TCP and UDP.
CO5: Ability to summarize and interpret the basic concepts of Application-Layer paradigms and standard client-server protocols.

LIST OF EXPERIMENTS COVERED:

1. Write a TCL script to simulate the network described below

Consider a small network with five nodes n0, n1, n2, n3, n4, forming a star topology. The node n4 is at the center. Node n0 is a TCP source, which transmits packets to node n3 (a TCP sink) through the node n4. Node n1 is another traffic source, and sends UDP packets to node n2 through n4. The duration of the simulation time is 10 seconds.
Write a Tcl script (in the text area below) to simulate this scenario. Once you have written your simulation script, click on the "Run" button to execute it. You can verify your code by clicking on the "View Solution" button at the bottom of the page.

2. Write a TCL script to simulate a file transfer with ns2

Consider a client and a server. The server is running a FTP application (over TCP). The client sends a request to download a file of size 10 MB from the server. Write a script to simulate this scenario. Let node #0 be the server and node #1 be the client. TCP packet size is 1500 B. Assume typical values for other parameters.

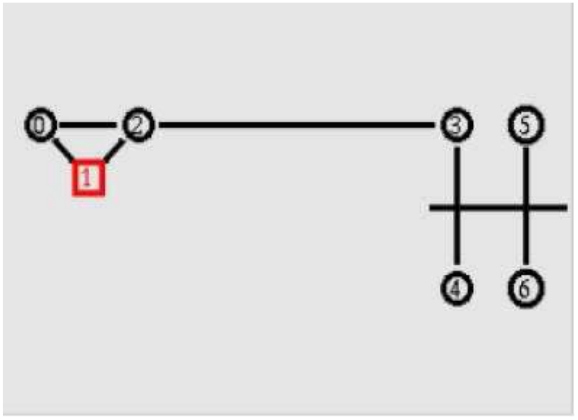
Note: This simulation require transfer of a **fixed** size file. Therefore, time required for the transfer would be constant for a given bandwidth of a link. To verify this, determine the time that would roughly be required for the transfer. Then look at the bottom of the trace file and verify whether there is any transmission beyond the time calculated. To verify that the client has downloaded the entire file, plot the "Bytes Received " curve for node #1. The y-axis is in Kbits. Convert it to MB and verify whether it approximates the specified file size. TCP headers would effectively increase the count of received bytes at node # 1.

3. Setting up a local area network with ns2

In this exercise you will be simulating a CSMA/CD based LAN with ns2. Consider the LAN with seven nodes to be an isolated one i.e. not connected to the Internet. Node # 0 in the LAN act as a UDP traffic source, and node # 6 is the destination node. Assume CBR traffic to be flowing between the nodes. The simulation lasts for 25 seconds.
In Ethernet a packet is broadcasted in the shared medium, and only the destination node accepts the packet. Other nodes simply drop it. What should be the number of hops a packet from node #0 to node # 6 travel? Verify this from the "Hop Count" plot.

Additional Task:
Suppose the above LAN is to be connected to the Internet. Add node # 7 into the network so that it act as the gateway. Connect node # 0 and # 7 with a 1 Mb wired link. Move the UDP source to node # 7. How the hop count should get affected in this case? Verify from the plot.

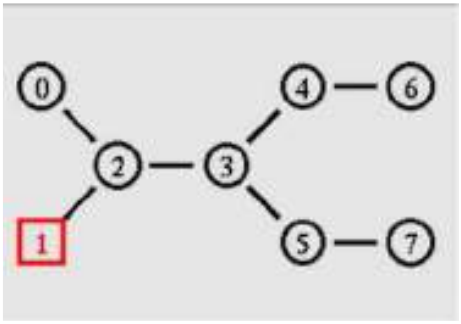
4. Simulating link errors
Consider the following network diagram



Here node # 2 act as a router. Any traffic to or from the LAN passes through it. Consider node #1 running a FTP server, and node # 5 is downloading a file of size 4 MB. However, the link between node # 2 and # 3 is fault. It drops packets with a fixed probability of 0.2. Implement a link error model to reflect this.
It may be noted here that the file download time will be more than the we had in exercise # 2 of experiment # 1. Try different values of the simulation time to ensure that the file has been entirely transferred. Is the plot of bytes received a linear curve or non-linear? Why? Presence of link errors cause one or more packets to be retransmitted. Verify this from the "Packet Retransmissions" plot.

5. Measuring Network Performances

Bottleneck in the network
Consider a dumbbell topology with eight nodes as shown as in the following figure.

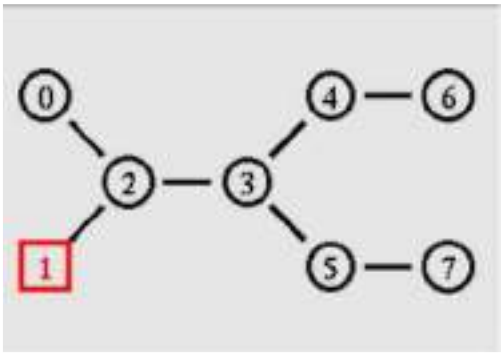


Consider nodes# 2 and 3 to be two routers connecting two different networks. When the bandwidth of the link 2-3 is much lower than the sum of bandwidths of the other links in the network, it act as a bottleneck.
Assume node # 0 running a FTP application (over TCP) and sending data to node # 6. Node # 1 is sending CBR data node # 7. Assume all the links except 2-3 has a bandwidth of 1 Mb, propagation delay of 10ms and queue type as DropTail. (All are duplex links).

Tasks:
• The link 2-3 has a propagation delay of 10 ms. Vary it's bandwidth from 0.5 Mb to 2.5 Mb in steps of 0.25Mb.
• Compute the throughput for node # 3 in each case
• Plot the throughput vs. bandwidth data in the "Custom Plot" section below
Based on the above plots, suggest what should be the recommended bandwidth of the link 2-3. Now, plot the end-to-end delay between nodes 0 and 6 for the above chosen values of link 2-3 bandwidth. Revisit your previous answer (i.e. optimum bandwidth of link 2-3) based on these graphs.

Measuring Network Performances

6. Bandwidth sharing between TCP and UDP
Consider the dumbbell topology from our previous exercise:



Node # 0 is a TCP source, and the corresponding sink is at node # 6. Node # 1 is a UDP source (CBR traffic) with a null agent attached to node # 7. These two traffic flows through the common link 2-3. The aim of this exercise is to examine how TCP and UDP share the bandwidth between themselves when the rate of CBR traffic is changed.
Set the TCP packet size to 1460 B. The UDP and CBR packet sizes are 1500 B. All the links in the network have same bandwidths (say, 4 Mb), delay and queue types.

Part 1:

• Set the initial rate of CBR traffic to 0.5 Mb. Run the simulation, and plot the "Bytes Received" by node #s 4 and 5 (sinks for TCP and UDP traffic)

• Now, increment the rate up to 4 Mb, the link bandwidth, in steps of 0.5 Mb. Run the simulation and plot the graphs again.

How does the graphs change after each run? In particular, what's the nature of the graphs when the rate of CBR traffic is 50% of the bandwidth?

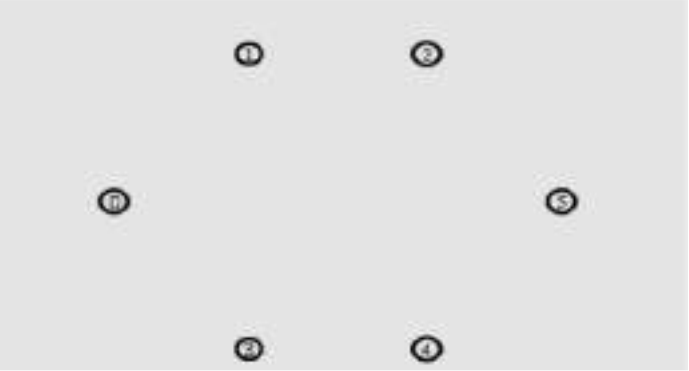
Part 2: Behaviour of UDP

• Reduce the bandwidth of the link 2-3 to say, 2 Mb. Repeat the above steps and observe the graphs in this case.

From the graphs plotted observe how UDP occupies a larger portion of the bandwidth. How Does the behaviour change for other variations of TCP (Newreno, Vegas)?

7. Write a TCL script to simulate the following scenario with ns2 simulator.

Consider six nodes, (as shown in the figure below) moving within a flat topology of 700m x 700m.



The initial positions of nodes are 0 (150,300) ,1 (300,500),2 (500,500),3 (300,100),4(500,100) and 5(650,300) respectively.A TCP connection is initiated between node 0 (source) and node 5 (destination) through node 3 and node 4 i.e the route is 0-3-4-5.

At time t = 3 seconds the FTP application runs over it.

After time t=4.0 sec, node 3 (300,100) moves towards node 1 (300,500) with a speed of 5.0m/sec

and after some time the path break, then the data transmit with a new path via node 1 and node 2 i.e the new route 0-1-2-5.

The simulation lasts for 60 secs. In the above said case both the route has equal cost.

Use DSR as the routing protocol and the IEEE 802.11 MAC protocol.

Now Analyze the trace file and determine when the use of second route commence, and

Plot the number of packets received by each node over the entire time duration of the simulation

8. Simulate a wired network and demonstrate Distance Vector Routing algorithm.

9. Simulate a network which will create congestion in the network. With the trace file created identify the points at which congestion occurs by writing sed / awk scripts. Also write a mechanism to correct/control the congestion.

Note : Network Free and open source software simulators like NS2 / NS3 could be used. If NS2 is used tcl scripting should be introduced . If NS3 is used c++ with python has to be introduced during first two or three weeks of the labs.

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. 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.

ADDITIONAL LEARNING SOURCES:

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

CO-PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	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

IV Semester

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**	
II	IV	Theory	04	04	00	00	52
Course No		**Course Title**				**Pre Requisites**	
MCA410		Python Programming				OOPS 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:

CO1: Design and apply a solution clearly, accurately in a program using python.

CO2: Comprehend and Apply knowledge in real time situational problems and think creatively about solutions.

CO3: Apply the best features of mathematics, engineering and natural sciences to program using python.

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

CO5: Demonstrate how to build and package python modules for reusability.

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:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	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

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	
II	IV	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA420	Data and Web Mining	DBMS

COURSE ASSESSMENT METHOD:

Internal Assessment [5 Events: 3 Written Tests, 2 Events] Marks: 50 [10* 5 Events].
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: Apply knowledge discovery techniques while mining the data; recognize & fixing the issues in data mining.
CO3: To apply the techniques of clustering, classification, association finding, feature selection and visualization of real world data.
CO4: Demonstrate the real world problem has a data mining solution.
CO5: Apply evaluation metrics to select data mining techniques.

TOPICS COVERED:

UNIT 1 - Introduction **10 Hours**

Data Mining, Functionalities, Data Cleaning, Data Integration and Transformation, Data Reduction. Data Mining Primitives, languages, and system Architectures, A Data Mining Query Language. Data Mining Applications, Trends in Data Mining.

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, Prediction and Cluster Analysis **12 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. What is Cluster Analysis? Types of data in Cluster Analysis: A Categorization of Major Clustering Methods. Partitioning Methods, Hierarchical Methods, Outliner Analysis.

UNIT 4 - Web Mining, Search and Link Analysis **10 Hours**

Text and Web Page pre-Processing, Inverted Index and its Compression, Latent Semantic Indexing, Web search, Meta Search: combining Multiple Rankings, Combination Using Similarity Scores,

Web Spamming, Link Analysis, Social Network Analysis Co-Citation and Bibliographic coupling, Page Rank HITS, Community discovery.

UNIT 5 - Social Network analysis, Mining Multimedia and World wide web 10 Hours

What is social network, Characteristics of social networks, Mining social networks, Similarity search in multimedia data, Multi dimensional analysis of multimedia data, classification and prediction of multimedia data, mining associations in multimedia data. Mining webpage layout structure, Mining multimedia data on the web, Automatic classification of web documents, Web usage mining.

TEXT BOOKS / REFERENCES:

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

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

Liu. B, "Web Data Mining, Exploring Hyperlinks, Contents and Usage Data", Springer, 2012.

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.

3.http://facweb.cs.depaul.edu/mobasher/classes/ect584/syllabus.html

4.https://www.cs.uic.edu/~liub/WebMiningBook.html

CO - PO MAPPING:

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

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	
II	IV	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA430	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: Comprehend the need and importance of software architectures.
CO2: Differentiate various architectural styles based on requirement.
CO3: Implement system qualities during architecture development for the application.
CO4: Apply pattern oriented architecture by understanding patterns and their descriptions.
CO5: Design and document the software architecture.

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:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	H	H	H	H	M	M	M	H	M
CO2	H	M	H	M	M	M	H	L	M	M	M	M
CO3	H	M	H	M	M	M	H	H	M	M	M	M
CO4	H	M	M	M	M	M	H	L	M	M	M	H
CO5	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	
II	IV	Theory	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA440	Cryptography and Network Security	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 able to:

CO 1: Explore the need for computer security concepts.

CO 2: Apply the principles and techniques of symmetric key encryption and public key encryption.

CO 3: Demonstrate the specifics of message authentication codes and hash algorithms.

CO 4: Analyze the facts of e-mail security and IP security evolution.

CO 5: Comprehend Web Security, Secure Electronic Transaction, Intruder detection and Firewalls.

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 and Key Management 11 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, Key Management, Diffie Hellman Key Exchange.

UNIT 3 - Cryptographic Hash Functions, Message Authentication Codes 12 Hours
Applications of Cryptographic Hash Functions, Message Authentication, Digital Signatures, Two Simple Hash Functions, Requirements and Security, Security Requirements for Cryptographic Hash Functions, Brute-Force Attacks, Cryptanalysis, 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 – Electronic Mail Security and IP Security 10 Hours
Pretty Good Privacy (PGP), S/MIME, IP Security Overview; IP Security Architecture; Authentication Header; Encapsulating Security Payload; Combining Security Associations.

UNIT 5 - User Authentication Protocols and Entity Authentication 10 Hours Web Security Considerations; Secure Socket Layer (SSL) and Transport Layer Security (TLS); Secure Electronic Transaction (SET), Intruders, Intrusion Detection, Firewall Design Principles- Characteristics, Types of Firewall and Firewall Configuration.

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:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	H	H	H	H	M	M	M	H	M
CO2	H	M	H	M	M	M	H	L	M	M	M	M
CO3	H	M	H	M	M	M	H	H	M	M	M	M
CO4	H	M	M	M	M	M	H	L	M	M	M	H
CO5	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	
II	IV	Theory (Stream)	05	04	00	01	65

Course No	Course Title	Pre Requisites
MCA 451	Artificial Intelligence and Robotics	--

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:** Find appropriate idealizations for converting real world problems into AI problems formulated using the appropriate search algorithm.
- **CO2:** Formulate and implement the appropriate search algorithms to find the solutions for real time and heuristics problems.
- **CO3:** Represent and debug knowledge in an appropriate first order logic representation with the understanding of the fundamentals of knowledge representation.
- **CO4:** Choose and Implement the appropriate algorithms for a real world supervised learning problem.
- **CO5:** Inculcate the basic knowledge of Robotics along with the Artificial Intelligence

TOPICS COVERED:

UNIT: 1 – Introduction to AI, Informed Search and Exploration **10 Hours**

Intelligent Agents: Agents and environment; Rationality; the nature of environment; the structure of agents. Problem-solving: Problem solving agents; Example problems; Searching for solution; uninformed search strategies.

Informed search strategies; Heuristic functions; On-line search agents and unknown environment.

UNIT: 2 - Constraint Satisfaction, Adversial Search, Logical Agents: **10 Hours**

Constraint satisfaction problems; Backtracking search for CSPs. Adversial search: Games; Optimal decisions in games; Alpha-Beta pruning.Knowledge-based agents; The wumpus world as an example world; Logic; propositional logic Reasoning patterns in propositional logic; Effective propositional inference; Agents based on propositional logic.

UNIT: 3 - First-Order Logic, Inference in First-Order Logic: **12 Hours**

Representation revisited; Syntax and semantics of first-order logic; Using first-order logic; Knowledge engineering in first-order logic. Propositional versus first-order inference; Unification and lifting; Forward chaining; Backward chaining; Resolution.

UNIT: 4 - Knowledge Representation and Learning, AI: Present and Future: 10 Hours

Ontological engineering; Categories and objects; Actions, situations, and events; Mental events and mental objects; The Internet shopping world; Reasoning systems for categories; Reasoning with default information; Truth maintenance systems.

Learning: Forms of Learning; Inductive learning; Learning decision trees; Ensemble learning; Computational learning theory. AI: Present and Future: Agent components; Agent architectures; Are we going in the right direction? What if AI does succeed? Game theory.

UNIT: 5 – Introduction to Robotics : 10 Hours

Introduction; Robot Hardware: sensors and Effectors; Robotic Perception: localization, mapping, other types of perception; Planning to Move: configuration space, cell decomposition methods and skeletonization methods; Planning uncertain movements: robust methods; Moving: dynamics and control, potential field control and reactive control; Robotic Software: architectures, subsumption architecture , three-layer architecture and robotic programming languages ; Application domains.

TUTORIALS:

1. Program to design tic-tac-toe game.
2. Program for breadth first and depth first search.
3. Program to N-Queeens Problem.
4. To implement max-min problem.
5. To implement simulated Annealing Algorithm.
6. Write a program to implement A* program.
7. To implement Hill-Climbing Algorithm.

Text Book / References:

Text Book:

1. Stuart Russel, Petr Norvig: Artificial Intelligence A Modern Approach, 2nd Edition, Pearson Education, 2003.

References:

1. Elaine Rich, Kevin Knight: Artificial Intelligence, 2nd Edition, Tata McGraw Hill, 1991.
2. Nils J. Nilsson: Principles of Artificial Intelligence, Elsevier, 1980.

CO-PO MAPPING:

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

Course Year	Course Semester	Course Type	Credits	Contact Hours/ Week			Total Hours/ Semester
				Theory	Laboratory	Tutorials	
II	IV	Theory [Stream]	05	03	00	02	65

Course No	Course Title	Pre Requisites
MCA452	NoSQL	RDBMS

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:** Demonstrate competency in describing how NoSQL databases differ from relational databases from a theoretical perspective.
- **CO2:** Demonstrate competency in designing NoSQL database management systems
- **CO3:** Use of a number of NoSQL databases to store and retrieve data and perform aggregation functions
- **CO4:** Demonstrate competency in selecting a particular NoSQL database for different applications.
- **CO5:** Execute various CRUD operations with MongoDB.

TOPICS COVERED:

UNIT 1 – An Overview and Characteristics of NoSQL **10 Hours**

Introduction to NoSQL : An Overview of NoSQL , Defining NoSQL, History, What NoSQL is and what it is not, Why NoSQL?, List of NoSQL Databases. Characteristics of NoSQL: Application, RDBMS approach, Challenges, NoSQL approach.

UNIT 2 – NoSQL Storage Types **12 Hours**

Modifying and managing NOSQL, Data stores, Indexing and ordering datasets (MongoDB/ CouchDB / Cassandra) NoSQL Storage Types : Storage types, Column-oriented databases, Document store, key-value store, graph store, multi-storage type databases, comparing the models.

UNIT 3 – Advantages and Drawbacks **10 Hours**

Advantages and Drawbacks : Transactional application, Computational application, Web-scale

application. Performing CURD operations : Creating records, accessing data, updating and deleting data.

UNIT 4 - Querying SQL **10 Hours**

Querying NoSQL stores : similarities between NoSQL and MongoDB query features. Managing data stores and managing evolutions.

UNIT 5 – Indexing and Ordering **10 Hours**

Indexing and ordering data sets: Essential concepts behind database index, indexing and ordering in MongoDB, indexing and ordering in CouchDB, Comparative Study of NoSQL Products Comparison: Technical comparison, Implementation language, Engine types, Speed, Features, Limits, Bulk operations, Bulk read, Bulk insert, Bulk update, Bulk delete, Query options.

TUTORIAL

Case Study

1. Application definition, Requirement analysis, Implementation using MongoDB, Features and constraints.
2. Database design, Database queries, Database modeling, Schema definition, Writing queries.
3. Queries for a single entity, simple result, Queries for a single entity, Aggregate, Queries for a one to one relationship.
4. Queries for a one to many relationship, Queries for a many to many relationship, Miscellaneous queries.
5. Pagination, Limiting items in an array in result set.
6. Plug-in and dynamic data support, Model refinements.
7. Reference using non-ID property, Demoralizations and document embedding.
8. Complete document embedding and Partial document embedding.
9. Bucketing, Cache document approach, Miscellaneous changes.

TEXT BOOKS / REFERENCES:

TEXT BOOKS :

1. Shanshank Tiwari “Professional NOSQL”, WROX Press, 2011
2. Pramod.J.Sadalage and Martin Fowler, “NoSQL Distilled : A Brief guide to the emerging world of polygot persistence”, Pearson Education corporation, I Edition,

2014.

Reference Books :

1. The definitive guide to MONGODB, The NOSQL Database for cloud and desktop computing, Apress 2010.

ADDITIONAL LEARNING SOURCES:

1. https://www.mongodb.com/nosql-explained
2. http://www.dbta.com/Editorial/Trends-and-Applications/NoSQL-for-the-Enterprise-80198.aspx
3. http://www.oracle.com/technetwork/database/databasetechnologies/nosqldb/overview/index.html

.

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	L	H	H	M	M	M	L	M	H	H
CO3	H	H	L	M	H	M	H	H	L	M	H	H
CO4	H	H	M	H	H	L	M	M	M	M	H	H
CO5	H	H	H	L	L	H	H	H	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	
II	IV	Theory (Stream)	05	03	00	02	65

Course No	Course Title	Pre Requisites
MCA453	Business Intelligence	ERP

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: Comprehend the basics and fundamentals of BI with its business and technical needs
CO2: Use the requirements and architectural framework of BI
CO3: Know and differentiate different components of the BI framework
CO4: Design BI concepts by understanding the requirement needs
CO5: Analyze and implement advanced BI techniques and analytics

TOPICS COVERED:

UNIT 1-The Business Demand for Data, Information, and Analytics 10 Hours

Just One Word:Data, Welcome to the Data Deluge, Data Volume, Variety, and Velocity, Taming the Analytics Deluge, The Importance of Analytics, Analytics Challenge, Analytics Strategy, Too Much Data, Too Little Information, The Difference Between Data and Information, The Role of BI in Creating Actionable Information, The Information Backbone, Data Capture versus Information Analysis, The Roles of BI and Operational Systems, Operational BI Blurs the Lines, Where Data Warehousing Fits in, The Five Cs of Data, Common Terminology from Our Perspective, References.

Justifying BI: Building the Business and Technical Case:Why Justification is Needed, Building the Business Case, Review Organization's Business Initiatives and Processes, Solicit BI Sponsorship, Enlist BI Stakeholders, Identify Business Processes Affected by BI, Document Business Benefits, Determine Business Value(Tangible Benefits), Business the Technical Case, Technology and Product Short Lists, Convincing Business People, Convincing the Technologists, Assessing Readiness, Data and Data Quality, Expertise and Experience, Organizational and Cultural Change, Financial and Resource Commitment, Creating a BI Road Map, Developing Scope, Preliminary Plan and Budget, Project Scope, Project Plan, Project Budget, Calculating Benefits and ROI, Obtaining Approval, Common Justification Pitfalls, Overzealous Business Sponsor, CIO is Sole Sponsor, Intangible or Too High-Level Benefits, Confusion Between BI Technology and Business Value.

UNIT 2-Defining Requirements-Business, Data and Quality 10 Hours

The Purpose of Defining Requirements, Goals, Deliverables, Roles, BI Team Participants, Business Participants, Other IT Participants, Defining Requirements Workflow, Business Requirements, Data(and Data Quality) Requirements,Functional Requirements, Regulatory/Compliance Requirements, Technical Requirements, Reverse Engineering(When Necessary), Putting It All Together, Prioritizing Requirements, Interviewing, Preparation for Interviews, Conducting the Interviews, Reviewing Interview Content, Interview Follow-ups, Documenting Requirements.

Architecture Framework:The Need for Architectural Blueprints, Architectural Framework, Information Architecture, Data Architecture, The Rise of the Enterprise Data Warehouse, Data Warehousing Replaces the Data Warehouse, Technical Architecture, Business Intelligence, Data Warehouse and BI Data Stores, Data

Integration, Source Systems, BI Technology Keeps Evolving, Product Architecture, Metadata, What is It?, What to do About It, Security and Privacy, Getting Started, Implementing the Plan, Avoiding Accidents with Architectural Planning, The Signs of Accidental Architectural Planning, Recovering from an Accidental Architecture, Do Not Obsess over the Architecture.

UNIT 3-Information Architecture **10 Hours**

The Purpose of an Information Architecture, Data Integration Framework, DIF Information Architecture, Data Preparation, Data Franchising, BI and Analytics, Data Management, Metadata Management, Operational BI versus Analytical BI, Shift All Reporting to the Application-Specific Environment, Shift All Reporting to the DW-Based BI Environment, Blend Application-Specific and DW BI Environments, Master Data Management, Identify the Data, Find the Problem Areas, Assess a Solution.

Data Architecture:The Purpose of a Data Architecture, History, Prehistory, In the Beginning, Data Warehousing Goes Public, The Data Mart, Multiple Data Marts, Operational Data Store(ODS), Federated DWs, BI Accidental Architecture, Hub-and-Spoke, Data Architectural Choices, Data Categories, Selecting a Data Architecture, The Same But Different, Analytical Data Architecture(ADA),Data Integration Workflow, Data Integration Workflow—Hub-and-Spoke, Data Workflow of the System of Integration(SOI), Data Workflow of the System of Analysis(SOA), Data Workflow—Rise of EDW Again, Operational Data Store, The Relational for an ODS, ODS Reexamined, ODS is Dead, Long Live ODS, References.

UNIT 4-Technology & Product Architectures **10 Hours**

Where are the Product and Vendor Names?, Evolution Not Revolution, Technology Platforms, Enterprise Applications, Data Management, Technology Architecture, Business Intelligence and Analytics, Information Access and Data Integration, Databases, Product and Technology Evaluations, BI Product Vendors, Dazed and Confused, Technology and Product Evaluations, Product Migration.

Business Intelligence Applications:BI Content Specifications, Revise BI Applications List, BI Personas, Casual Consumers, Analyst, Power Users, Data Scientists, BI Design Layout—Best Practices, Focus on the Purpose, Design Layout, Data Design for self-Services BI, The Last Data Preparation Step, When Inconsistency is Reintroduced, OLAP Cubes and In-Memory Columnar Databases, Matching types of analysis to Visualizations, Comparative Analysis, Time-series or Trending Analysis, Contribution Analysis, Correlation Analysis, Geographic Data, Distribution Analysis.

UNIT 5-BI Design and Development **12 Hours**

BI Design, BI User Interface(UI) Standards, Create Privacy, Security and Access Standards, Designing Each BI Application, BI Development, Prototyping Lifecycle, BI Application Development Tasks, BI Application Testing.

Advanced Analytics:Advanced Analytics Overview and Background, The Window to the Future, Don't Ignore the Past, Advanced Analytics in Action, Predictive Analytics and Data Mining, Setting Up a Predictive Analytics or Data Mining Program, Tasks for Developing and Using Models, Selecting Tools, Architecture for Predictive Analytics and Data Mining, Techniques for Predictive Analytics and Data Mining, Resources and Skills, Roadblocks to Success, Analytical Sandboxes and Hubs, Analytical Sandboxes, Analytical Hubs, Hub and Sandbox Design Principles, Hub and Sandbox Architecture Options, Advice for Hubs and Sandboxes, Big Data Analytics, Scope, The Program, Hybrid Architecture, The Big Data Team, Big Data Analytics Worst Practices, Data Visualization, Why Data Visualization is Needed, Why Data Visualization is Not, References.

TUTORIALS:

1. Case study on requirement specification.
2. Defining a requirement workflow.
3. Writing technical requirement specification.
4. Prioritizing the requirements.
5. Writing a BIE roadmap.
6. Writing & Defining scope objectives & outcomes.
7. Writing a plan & budget.
8. Writing Data Management techniques.
9. Creating data marts & operational data stores.
10. Creating OLAP Cubes.

TEXT BOOKS / REFERENCES:

Text books:

1. **"**Business Intelligence Guidebook: From Data Integration to Analytics" Book by Rick Sherman, 1st Edition, 2014.

Reference books:

1. Business Intelligence Roadmap: The Complete Project Lifecycle for Decision-Support Applications by Larissa T. Moss and ShakuAtre, February 25th 2003.
2. Successful Business Intelligence, Second Edition: Unlock the Value of BI & Big DataHardcover– Import, 1 Nov 2013.
3. Predictive Analytics: The Power to Predict Who Will Click, Buy, Lie, or Die, Revised and UpdatedPaperback– 22 Feb 2016.

ADDITIONAL LEARNING SOURCES:

1. https://thebipalace.com/
2. https://www.analyticsvidhya.com/learning-paths-data-science-business-analytics-business-intelligence-big-data/tableau-learning-path/

CO - PO MAPPING:

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

JSS Science and Technology University, Mysuru Department of Master of Computer Applications [MCA]					
Course	**Course**	**Course**	**Credit**	**Contact Hours/ Week**	**Total**

Year	Semester	Type	s	Theory	Laboratory	Tutorials	Hours/ Semester
II	IV	Laboratory	01	00	03	00	48

Course No	Course Title	Pre Requisites
MCA41L	Python Programming Laboratory	OOPS 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 able to:

CO1: Design and apply a solution clearly, accurately in a program using python.

CO2: Comprehend and Apply knowledge in real time situational problems and think creatively about solutions.

CO3: Apply the best features of mathematics, engineering and natural sciences to program using python.

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

CO5: Demonstrate how to build and package python modules for reusability.

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:

1. Timothy A. Budd: Exploring Python, Tata McGraw-Hill, 2011.
2. 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.
3. 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	PO8	PO9	PO10	PO11	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

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	
II	IV	Laboratory	01	00	03	00	48

Course No	Course Title	Pre Requisites
MCA42L	Data and Web Mining Laboratory	RDBMS, ADA

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, the student will be able to:

CO1: Interpret the basic concepts, principles and techniques of data mining.

CO2: Apply knowledge discovery techniques while mining the data; recognize & fixing the issues in data mining.

CO3: To apply the techniques of clustering, classification, association finding, feature selection and visualization of real world data.

CO4: Demonstrate the real world problem has a data mining solution.

CO5: Apply evaluation metrics to select data mining techniques.

LIST OF CASE STUDIES COVERED:

The following are the list of algorithms where, the students are going to trace the steps of the algorithm using advanced programming and demonstrate the flow of the algorithm.

1. Apriori Algorithm.
2. Partition algorithm.
3. DIC Algorithm.
4. FP-Tree Algorithm.
5. Update algorithm.
6. Border algorithm.
7. Decision Tree Algorithm.
8. K-medoids Algorithm.
9. K-means Algorithm.
10. Generalized Sequential Patterns.
11. KNN Algorithm.
12. K-mean Clustering Algorithm.
13. Apriori Influencial Algorithm.

TEXT BOOKS / REFERENCES:

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

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

3. Liu. B, "Web Data Mining, Exploring Hyperlinks, Contents and Usage Data", Springer, 2012.

ADDITIONAL LEARNING SOURCES:

3.web.cse.ohio-state.edu/~srini/674/part1.ppt.

4.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	M	M	H	M	M	M
CO2	H	M	M	H	M	M	H	M	M	M	L	L
CO3	M	M	M	H	M	H	H	M	H	H	L	L
CO4	H	M	M	H	M	M	M	M	M	M	L	L
CO5	H	H	H	H	M	M	H	H	M	M	L	L

JSS Science and Technology University, Mysuru
Department of Mater of Computer Applications [MCA]

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
MCA43L	Applications Laboratory	Programming Languages, Software Engineering

COURSE ASSESSMENT METHOD:

1. Internal Assessment [3 Events: Exploration, Implementation and Presentation]
   Total Marks: 50 [15+15+20].

COURSE OUTCOMES:

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

CO1: Exploration of different application domains.
CO2: Create small applications using previous knowledge.
CO3: Follow standards of Software engineering.
CO4: Implement application using platforms like Java, PHP, Python etc.
CO5: Present and document the application implementation by following standards

TOPICS COVERED:

Students are expected to build an application by using the concepts learned earlier like

- Standard procedures of software engineering,
- Languages such as Java, Python, PHP etc.
- Database like MySQL, Oracle, MS SQL etc. and
- Their areas of interest like machine Learning, Networking, Business applications etc.

ADDITIONAL LEARNING SOURCES:

1. http://www.ibm.com/developerworks/websphere/library/techarticles/0306_perks/perks2.html
2. https://hedleyproctor.com/software-development-best-practices/
3. http://www.comentum.com/guide-to-web-application-development.html
4. http://www.tutorialspoint.com/developers_best_practices/
5. https://www.sans.org/reading-room/whitepapers/application/framework-secure-application-design-development-842
6. https://msdn.microsoft.com/en-us/library/aa260844(v=vs.60).aspx
7. https://www.owasp.org/index.php/OWASP_Secure_Coding_Practices_-Quick_Reference_Guide
8. http://www.computerworld.com/article/2572077/app-development/best-practices-for-software-development-projects.html

CO-PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	H	H	H	H	H	H	M	M	M	H	M
CO2	H	M	H	M	M	M	H	L	M	M	M	L
CO3	H	M	H	M	M	M	H	L	M	M	M	L
CO4	H	M	M	M	M	M	H	L	M	M	M	L
CO5	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	
II	IV	Laboratory	01	00	03	00	48

Course No	Course Title	Pre Requisites
MCA44L	Cryptography and Network Security Laboratory	Computer Networks

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:

CO 1: Explore the need for computer security concepts.

CO 2: Apply the principles and techniques of symmetric key encryption and public key encryption.

CO 3: Demonstrate the specifics of message authentication codes and hash algorithms.

CO 4: Analyze the facts of e-mail security and IP security evolution.

CO 5: Comprehend Web Security, Secure Electronic Transaction, Intruder detection and Firewalls.

LIST OF EXPERIMENTS COVERED:

1. Implementation of Caesar Cipher technique
2. Implementation of Shift or Additive Cipher Technique
3. Implementation of Poly-alphabetic Cipher Technique
4. Implement the Playfair Cipher
5. Implement the Pure Transposition Cipher
6. Implement DES Encryption and Decryption
7. Implement the AES Encryption and decryption
8. Implement RSA Encryption Algorithm
9. Implementation of Diffie Hellman Key Exchange
10. Implementation of Hash Functions

11. Implementation of Digital Signature
12. Test and understand the function of different Firewalls.

Note: Students may use any programming languages like C, C++ or Java etc. for implementation of experiments

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:

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

V Semester

JSS Science and Technology University, Mysuru
Department of Mater of Computer Applications [MCA]

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	.NET Technologies	OOPS,Java

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: Develop application using the concept of .NET framework and basics of C# .NET.
CO2: Create server side applications using C#.NET.
CO3: Develop web applications using the ASP.NET.
CO4: Comprehend 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 –Introduction to .NET and Basics of C#.NET 12 Hours

The .NET Framework: The Evolution of Web Development-HTML and HTML Forms, Server-Side Programming, Client-Side Programming. The .NET Framework - C#, VB, and the .NET Languages, The Common Language Runtime, The .NET Class Library, Visual Studio. The .NET Languages, C# Language Basics, Case Sensitivity, Commenting, Statement Termination, Blocks, Variables and Data Types-Assignment and Initializes, Strings and Escaped Characters, Arrays, Enumerations, Variable Operations- Advanced Math, Type Conversions. Object-Based Manipulation- String, DateTime and TimeSpan Types, The Array Type. Conditional Logic-The if, switch Statement. Loops- The for, foreach, while loop, Methods-Parameters, Method Overloading, Optional and Named Parameters, Delegates.

UNIT 2 – C#.NET Types, Objects, and Namespaces 10 Hours

The Basics About Classes-Static Members, A Simple Class, Building a Basic Class-Creating an Object, Adding

Properties, Automatic Properties, Adding a Method, Adding a Constructor, Adding an Event, Testing the Product Class. Value Types and Reference Types-Assignment Operations, Equality Testing, Passing Parameters by Reference and by Value, Reviewing .NET Types. Understanding Namespaces and Assemblies-Using Namespaces, Importing Namespaces, Assemblies. Advanced Class Programming-Inheritance, Static Members, Casting Objects, Partial Classes, Generics.

UNIT 3 - Developing ASP.NET Applications **10 Hours**

Visual Studio: Creating Websites-Creating an Empty Web Application, Websites and Web Projects, The Hidden Solution Files, The Solution Explorer, Adding Web Forms, Designing a Web Page-Adding Web Controls, The Properties Window, and The Anatomy of a Web Form-The Web Form Markup, The Page Directive, The Doctype, The Essentials of XHTML. Writing Code-The Code-Behind Class, Adding Event Handlers, Outlining, IntelliSense, Code Formatting and Coloring, Visual Studio Debugging-The Visual Studio Web Server, Single-Step Debugging, Variable Watches, The Anatomy of an ASP.NET Application-ASP.NET File Types, ASP.NET Application Directories. Introducing Server Controls-HTML Server Controls, Converting an HTML Page to an ASP.NET Page, View State, The HTML Control Classes, Adding the Currency Converter Code, Event Handling, Error Handling

UNIT 4 – ASP.NET Web Form Basics, State Management & Error Handling **10 Hours**

ASP.NET Configuration- The web.config File, Nested Configuration, Storing Custom Settings in the web.config File, The Website Administration Tool (WAT), Web Controls-Basic Web Control Classes, The Web Control Tags, Web Control Classes, List Controls, Table Controls, Web Control Events and AutoPostBack, A Simple Web Page. The Problem of State-View State, Transferring Information Between Pages, Cookies, Session State, Session State Configuration, Application State, An Overview of State Management Choices. Error Handling, Logging, and Tracing-Common Errors, Exception Handling-The Exception Class, The Exception Chain, Handling Exceptions, Throwing Your Own Exceptions, Logging Exceptions, Page Tracing.

UNIT 5 – ADO.NET Fundamentals **10 Hours**

Understanding Databases, Configuring Your Database-SQL Server Express, Browsing and Modifying Databases in Visual Studio, The *sqlcmd* Command-Line Tool. SQL Basics - Running Queries in Visual Studio, The Select, Update, Insert, Delete statement. The Data Provider Model: Direct Data Access-Creating a Connection, The Select Command, The DataReader, Putting It All Together, Updating Data. Disconnected Data Access-Selecting Disconnected Data, Selecting Multiple Tables, Defining Relationships. Introducing Data Binding-Types of ASP.NET Data Binding, How Data Binding Works, Single-Value Data Binding, Repeated-Value Data Binding, Data Source Controls.

TEXT BOOKS / REFERENCES:

Text books:

1. Matthew MacDonald. Beginning ASP.NET 4 in C# 2010, APRESS, 2010

Reference books:

2. Joseph Mayo. Visual studio 2010 - A beginners guide – BPB Publications 2010
3. Greg Buczek: ASP.Net Developer's Guide, Tata McGraw Hill Edition 4th Edition, 2005.
4. Pro ASP.NET 4 in C# 2010, MacDonald and Freeman

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:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	H	H	H	H	H	H	M	M	M	H	M
CO2	H	M	H	M	M	M	H	L	M	M	M	L
CO3	H	M	H	M	M	M	H	L	M	M	M	L
CO4	H	M	M	M	M	M	H	L	M	M	M	L
CO5	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	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA520	Mobile Applications	OOPS, 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 to:

CO1: Gain broad understanding of the discipline of Mobile Application Development using J2ME Technology.

CO2: Develop User Interface for a J2ME application

CO3: Manage data on both service-side components and client-side applications and Address Portability and Compatibility issues between PDA'S and Cell phones.

CO4: Implement the design using Android SDK.

CO5: Implement the design using Objective C and iOS

TOPICS COVERED:

Unit 1 - J2ME Overview, Architecture and Development Environment 12 Hours

Java 2 micro edition and the world of java, inside J2ME, J2ME and wireless devices. small computing technology: wireless technology , radio data networks, microwave technology, mobile radio Networks, messaging, personal digital assistants.
J2ME architecture, small computing device requirements, run time environment, midlet programming, java language for J2ME, J2ME software development kits, hello world J2ME style, multiple midlets in a midlet suite, J2ME wireless toolkit.

Unit 2 - J2ME Best Practices and Patterns 10 Hours

The reality of working in a J2ME world, best practices commands, items, and event processing: J2ME user interfaces , display class, the palm OS emulator, C command class, item class, exception handling. high level display screens: screen class, alert class, form class, item class, list class, text box class, ticker class. low-level display canvas:
The Canvas, User Interactions, Graphics,Clipping Regions, Animation.

Unit 3 - Record Management System 10 Hours

Record storage, writing and reading records, record enumeration, sorting records, searching records, record listener. JDBC objects: The concept of JDBC, JDBC driver types, JDBC packages, overview of the JDBC process, database connection, statement objects, result set, transaction processing, metadata, data types, exception.

Unit 4 - Technology-I Android-12 10 Hours

Introduction – establishing the development environment – android architecture – activities and views – interacting with UI – persisting data using SQLite – packaging and deployment – Interaction with server side applications – Using Google Maps, GPS and Wifi – Integration with social media applications.

Unit 5 - Technology-II IOS-12 **10 Hours**

Introduction to objective C – iOS features – UI implementation – Touch frameworks – Data persistence using Core Data and SQLite – Location aware applications using Core Location and Map Kit – Integrating calendar and address book with social media application – Using Wifi - iPhone marketplace.

TEXT BOOKS / REFERENCES:

Text Books:

1. James Keogh , J2ME The Complete Reference , Tata McGrawHill.
2. Charlie Collins, Michael Galpin and Matthias Kappler, "Android in Practice", DreamTech, 2012.
3. David Mark, Jack Nutting, Jeff LaMarche and Frederic Olsson, "Beginning iOS.

Reference Books:

1. Michael Juntao Yuan, Enterprise J2ME, Developing Mobile Java Applications Pearson Education , 2011.
2. Sing Li, Jonathan B. Knudsen, Beginning J2ME: From Novice to Professional, Third Edition, Apress, 2015.
3. Development: Exploring the iOS SDK", Apress, 2013.

ADDITIONAL LEARNING SOURCES:

1. http://developer.android.com/develop/index.html.

CO-PO MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	H	L	L	M	L	M	M	H	H
CO2	H	H	H	H	L	L	M	L	M	L	H	H
CO3	H	H	H	H	L	L	M	L	M	L	H	H
CO4	H	H	H	H	L	L	M	L	M	M	H	H
CO5	H	H	H	H	L	L	M	L	M	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	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA530	Software Testing and Practices	Software Engineering

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: Gain knowledge on basics of Software Testing, Test case selection and creation.
CO2: Illustrate various perspectives of testing with examples.
CO3: Use by differentiating boundary value testing, Equivalence class testing, Decision table based testing.
CO4: Implement Path testing and Data flow testing based on the requirements
CO5: Comprehend different levels of testing, Integration testing and Fault based testing.

TOPICS COVERED:

UNIT 1- Basics of Software Testing 12 Hours

Humans, Errors and Testing, Software Quality; Requirements, Behavior and Correctness, Correctness Vs Reliability; Testing and Debugging; Test Metrics; Software and Hardware Testing; Testing and Verification; Defect Management; Execution History;Test Generation Strategies; Static Testing; Test Generation from Predicates.

Basic Principles, Test case selection and Adequacy

Sensitivity, Redundancy, Restriction, Partition, Visibility and Feedback, Test Specification and cases, Adequacy Criteria, Comparing Criteria

UNIT 2- A perspective on Testing, Examples 8 Hours

Basic definitions, Test cases, Insights from a Venn diagram, Identifying test cases, Error and fault taxonomies, Level of testing, Examples: Generalized pseudocode, The triangle problem,theNextDate function, The commission problem, The SATM (Simple Automation Teller Machine) problem, The currency converter, Saturn windshield wiper.

UNIT 3 - Boundary value, Equivalence class and Decision table based testing 8 Hours

Boundary value analysis, Robustness testing, Worst-case testing, special value testing, Examples, Random testing, Equivalence classes, Equivalence test cases for triangle problem, Next Date function and commission problem, Guidelines and observations, Decision tables, Test cases for triangle problem.

UNIT 4 - Path Testing, Data flow testing, Levels and Integration Testing 12 Hours

DD Paths, Test coverage metrics, Basis path testing, guidelines and observations, Definition
Use testing, Slice based testing, Guidelines and observations. Traditional view of testing levels, Alternative life cycle models, the SATM systems, separating integration and system testing, Guidelines and observations.

UNIT 5 - Fault Based Testing 12 Hours

Assumptions in fault-based testing, Mutation Analysis, Fault-based Adequacy Criteria; Variations on mutation Analysis; From Test case specification to Test Cases, Scaffolding, Generic vs specific Scaffolding, Test Oracles, Self checks as oracles, Capture and Replay.

Agile Testing

Definition and description, how is it different from traditional testing, ten principals for testers, business-facing the test that support the testing.

TEXT BOOKS / REFERENCES:

TEXT BOOKS :

1. AdithyaP.Mathur " Foundations of Software Testing – Fundamental Algorithms and Techniques", Pearson Education India, 2011
2. MauroPezze, Michael Young, Software testing and Analysis- Process, Principles and Techniques", Wiley India, 2012
3. Paul C Jourgensen, "Software Testing A Craftmans Approach", Aueredach publications, 3rd edition, 2011
4. Lisa Crisping, Janet Gregory, "Agile Testing : A Practical Guide for Testers and Agile Team", The Addison Wesley Signature Series, 2009.

REFERENCE BOOKS:

1. KshirasagaraNaik, PriyadarshiTripathy: Software Testing and Quality Assurance, Wiley India 2012
2. M.G.Limaye: Software Testing-Principels, Techniques and Tools – McGrawHill, 2009
3. Brain Marick: The Craft of Software Testing, Pearson Education India, 2008
4. Ron Patton: Software Testing, 2nd Edition, Pearson Education, India, 2013

ADDITIONAL LEARNING SOURCES:

1. http://www.softwaretestinghelp.com/agile-scrum-methodology-for-development-and-testing/
2. http://crbtech.in/Testing/agile-model-software-testing/
3. https://www.getzephyr.com/**test**-management/**agile**-**model**-in-**software**-**testing**
4. http://www.mountaingoatsoftware.com/
5. http://www.testingexperience.com/
6. http://www.infoq.com/

http://www.qasymphony.com/

.

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	L	H	M	H	M	H	M	H	M	H	L
CO4	H	L	H	M	H	M	L	M	H	M	H	M
CO5	H	M	H	L	H	H	L	L	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	04	04	00	00	52

Course No	Course Title	Pre Requisites
MCA540	Cloud Computing	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, the student will be able to:

CO1: Interpret the basic concepts, principles and techniques of data mining.

CO2: Apply knowledge discovery techniques while mining the data; recognize & fixing the issues in data mining.

CO3: To apply the techniques of clustering, classification, association finding, feature selection and visualization of real world data.

CO4: Demonstrate the real world problem has a data mining solution.

CO5: Apply evaluation metrics to select data mining techniques.

Topics Covered:

UNIT 1- Introduction to Cloud Computing **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 **10 Hours**

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: Capacity Planning, Defining Baseline and Metrics, Network Capacity, Scaling.

UNIT 3 - Exploring Platform as a Service **10 Hours**

Defining Services, Using PaaS Application Frameworks Using Google Web Services: Exploring Google Applications, Surveying the Google Application Portfolio, Exploring the Google Toolkit, Working with the Google App Engine. Managing the Cloud: Administrating the Clouds, Cloud Management Products, Emerging Cloud Management Standards.

UNIT 4 - Understanding Cloud Security **10 Hours**

Securing the cloud, Securing data, Establishing identity and Presence. 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.

UNIT 5: Using Webmail Services **10 Hours**

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

Text Books/ References:

Text Books:

1. Barrie Sosinsky "Cloud Computing Bible" 2011 by Wiley Publishing, Inc.

Reference Books:

1. Cloud Computing Principles and Paradigms by Rajkumar Buyya 2011, Published by John Wiley & Sons
2. Cloud Computing Theory and Practice by Dan C.Marinescu, 2013, Published by Morgan Kaufmann.

Additional Resource :

1. https://cloudacademy.com/ebooks
2. www.freebookcentre.net › Networking Books

CO-PO MAPPING

	P01	PO2	PO	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
C01	H	H	H	L	M	H	H	L	H	L	L	M
CO2	M	M	M	M	M	H	H	L	H	M	L	M
C03	L	M	M	H	H	H	M	H	M	H	M	M
C04	L	H	M	M	H	M	M	H	M	H	H	M
C05	H	H	M	H	L	L	M	M	M	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 (Stream)	05	03	00	02	65

Course No	Course Title	Pre Requisites
MCA551	Soft Computing	CVPR/AIR

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: Identify and describe soft computing techniques and their roles in building intelligent machines.
CO2: Recognize the feasibility of applying a soft computing methodology for a particular problem.
CO3: Apply fuzzy logic and reasoning to handle uncertainty and solve engineering problems.
CO4: Apply genetic algorithms to combinatorial optimization problems.
CO5: Apply neural networks to pattern classification and regression problems .

TOPICS COVERED:

UNIT 1 - Introduction **10 Hours**

Artificial neural network: Introduction, characteristics- learning methods – taxonomy – Evolution of neural networks- basic models – important technologies – applications. Fuzzy logic: Introduction – crisp sets- fuzzy sets – crisp relations and fuzzy relations: Cartesian product of relation – classical relation, fuzzy relations, tolerance and equivalence relations, non-iterative fuzzy sets. Genetic algorithm- Introduction – biological background – traditional optimization and search techniques – Genetic basic concepts.

UNIT 2 - Neural Networks **10 Hours**

McCulloch-Pitts neuron – linear separability – hebb network – supervised learning network: perceptron networks – adaptive linear neuron, multiple adaptive linear neuron, BPN, RBF, TDNN- associative memory network: auto-associative memory network, hetero-associative memory network, BAM, hopfield networks, iterative autoassociative memory network & iterative associative memory network – unsupervised learning networks: Kohonenself organizing feature maps, LVQ – CP networks, ART network.

UNIT 3 - Fuzzy Logic **10 Hours**

Membership functions: features, fuzzification, methods of membership value assignments- Defuzzification: lambda cuts – methods – fuzzy arithmetic and fuzzy measures: fuzzy arithmetic – extension principle – fuzzy measures – measures of fuzziness -fuzzy integrals – fuzzy rule base and approximate reasoning : truth values and tables, fuzzy propositions, formation of rules-decomposition of rules, aggregation of fuzzy rules, fuzzy reasoning-fuzzy inference systems-overview of fuzzy expert system-fuzzy decision making.

UNIT 4 – Genetic Algorithm **10 Hours**

Genetic algorithm and search space – general genetic algorithm – operators – Generational cycle – stopping condition – constraints – classification genetic programming – multilevel optimization – real life problem-advances in GA.

UNIT 5 – Hybrid Soft computing Techniques and Applications **12 Hours**

Neuro-fuzzy hybrid systems – genetic neuro hybrid systems – genetic fuzzy hybrid and fuzzy genetic hybrid systems – simplified fuzzy ARTMAP – Applications: A fusion approach of multispectral images with SAR, optimization of traveling salesman problem using genetic algorithm approach, soft computing based hybrid fuzzy controllers.

TUTORIALS:

Solving simple programs using MATLAB in the following areas:

1. FUZZY Logic techniques.
2. Neural networks techniques.
3. Genetic Algorithm techniques.
4. Hybrid Soft Computing Techniques.

TEXT BOOKS / REFERENCES:

Text books:

1. J.S.R.Jang, C.T. Sun and E.Mizutani, "Neuro-Fuzzy and Soft Computing", PHI / Pearson Education,2004.
2. S.N.Sivanandam and S.N.Deepa, "Principles of Soft Computing", Wiley India Pvt Ltd, 2011.

Reference books:

1. S.Rajasekaran and G.A.Vijayalakshmi Pai, "Neural Networks, Fuzzy Logic and Genetic Algorithm: Synthesis & Applications", Prentice-Hall of India Pvt. Ltd., 2006.

2. David E. Goldberg, "Genetic Algorithm in Search Optimization and Machine Learning" Pearson Education India, 2013.

ADDITIONAL LEARNING SOURCES:

1. http://www.myreaders.info/html/soft_computing.html

2. http://www.soft-computing.de/

CO - PO MAPPING:

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

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 (Streams)	05	03	00	02	65

Course No	Course Title	Pre Requisites
MCA552	Big Data Analytics	Data Warehousing and No SQL

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: Demonstrate the knowledge, significance structure and sources of Big Data.
CO2: Ability to think critically in making decisions based on data analytics, specific to Big Data.
CO3: Apply the technical skills in predicative and perspective modelling to support business decisions.
CO4: Comprehend decision tools and techniques for data streaming using various algorithms.
CO5: Demonstrate the Knowledge gained on mining social network data.

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 – cognos – microstrategy - pentaho. Analysis approaches – statistical significance – business approaches – analytic innovation – traditional approaches – iterative

UNIT 3 - Mining Data Streams **10 Hours**

Introduction to streams concepts, stream data model and architecture, stream computing, sampling data in a stream, filtering streams, counting distinct elements in a stream, estimating moments, counting oneness in a window, decaying window, realtime analytics platform(rtap) applications, case studies, real time sentiment analysis, stock market predictions.

UNIT 4 - Frequent Item sets and Clustering **10 Hours**

Mining frequent itemsets - market based model – apriori algorithm – handling large data sets in main memory – limited pass algorithm – counting frequent itemsets in a stream – clustering techniques – hierarchical – k- means – clustering high dimensional data – clique and proclus – frequent pattern based clustering methods – clustering in non-euclidean space – clustering for streams and parallelism.

UNIT 5 : Frameworks and Visualization **10 Hours**

Mapreduce – hadoop, hive, mapr – sharding – nosql databases - s3 - hadoop distributed file systems – visualizations - visual data analysis techniques, interaction techniques; systems and applications.

TUTORIALS

Case Studies:

1. Medicare and Medicaid Services : Integrity of health care data and secure payment processing.
2. Tesco PLC.
3. American Express Co.
4. Mobile Telecom Harnesses Big Data with Combined Actuate and Hadoop Solution.
5. Re-engineering a Telecom Market Share Analytical Application.
6. Telco Case Study: Vodafone and Argyle Data on using big data to combat fraud.
7. MTS India relies on HP Vertica in a highly competitive telecom market.
8. Mclaren's Formula One racing team : real time car sensor data during car races.

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. Anand Rajaraman and Jeffrey David Ullman, Mining of Massive Datasets, Cambridge University Press, 2014

Reference books:

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, 2012.
4. Glenn J. Myatt, Making Sense of Data, John Wiley & Sons, Pete Warden, Big Data Glossary, O"Reilly.

5. 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:

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

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 (Stream)	05	03	00	02	65

Course No	Course Title	Pre Requisites
MCA553	Software Project Management and Practices	MIS,SE,ST,ERP,BI

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: Comprehend software project management basics and approaches
CO2: Analyze different project management initiation techniques
CO3: Apply proper project planning and scheduling techniques
CO4: Execute software projects with efficient control mechanisms
CO5: Decide on closure of projects using standard and agile methodologies

TOPICS COVERED:

UNIT 1 -Software Project Basics **10 Hours**

Introduction, Types of Software Projects, Classifications of Software Projects, Based on Software Development Life Cycle, Approach Driven, Maintenance, Web Application, Agile Development, Conclusion
Approaches to Software Project Management:Alignment of Software Engineering Methodology with Project Management Methodology, The Ad Hoc Methods-Based Approach, The Process-Driven Approach, So, What Is the Right Approach?, The Ad Hoc Approach, The Process-Driven Approach, But Is a Process-Driven Approach the Right Choice?, In a Process-Driven Approach: What Process and How Much?
Software Project Acquisition:From an External Client, The Request for Proposal, The Proposal, Negotiation, Contract Acceptance, From an Internal Client, The Feasibility Study, Preparing the Proposal, Finalizing the Proposal, Reference.

UNIT2 -Software Project Initiation **10 Hours**

Introduction, Initiation Activities, Project Management Office-Level Activities, Identifying the Software Project Manager, Preparing/Handing Over the Project Dossier to the Software Project Manager, Coordinating Allocation of Project Resources, Assisting the Software Project Manager in Obtaining Necessary Service Level, Agreements from Departments in the Organization, Assisting the Software Project Manager with the Project Kickoff Meeting, Software Project Manager-Level Activities, Ensuring that Project Specifications Are Complete,Reviewing Estimates and Revisions/Updates of Estimates, Identifying Necessary Resources and Raising Requests, Preparing Project Plans, Setting Up the Development Environment,Arranging for Project-Specific Skill Training, Organizing the Project Team, Training the Project Team on the Project Plans, Conducting a Project Kickoff Meeting, Arranging for a Phase-End Audit, Common Pitfalls in Project Initiation, Identifying the Wrong Software Project Mana, Identifying Inappropriate Resources, Incurring Delays in Software Project Initiation Activities.

UNIT3 -Software Project Planning **10 Hours**

Introduction, Planning Defined, Plans Prepared in Software Project Management, The Project Management Plan, Resources, Skill Sets, Computer Systems, Project Management Method,The Configuration Management Plan, Naming Conventions, Change Management, The Quality Assurance Plan, The Schedule Plan, The Induction Training Plan, The Risk Management Plan, The Build Plan, The Deployment Plan, The User Training Plan,The Handover Plan, The Software Maintenance Plan, The Documentation Plan, Roles in Planning.
Scheduling:Introduction, The Initial Work Breakdown Structure, A Work Breakdown Structure with Predecessors Defined, A Work Breakdown Structure with Initial Dates, A Work Breakdown Structure with Resource Allocation, Scheduling in Practice, Graphic Representation of a Schedule

UNIT 4 -Software Project Execution **10 Hours**

Introduction, Work Management, Work Registers, De-allocation, Configuration Management,Information Artifacts, Code Artifacts, Configuration Registers, Configuration Management Tools, Quality Management, Verification Techniques, Validation Techniques, Product Testing,Allocation of Quality Assurance Activities, Productivity Management, Stakeholder Expectations Management, Product Integration Management.
Software Project Execution Control:Introduction, Aspects of Control in Project Execution, Scope Control, Cost Control, Schedule/Progress Control, Quality Control, Effort Control, Productivity Monitoring, Control Mechanisms, Progress Assessment: Earned Value Analysis.

UNIT 5 - Software Project Closure **12 Hours**

Introduction, Identifying Reusable Code Components, Documenting the Best Practices,Documenting the Lessons Learned, Collecting/Deriving and Depositing the Final Project Metrics in the Organizational, Knowledge Repository, Conducting Knowledge-Sharing Meetings with Peer Software Project Managers, Depositing Project Records with the Project Management Office, Depositing Code Artifacts in the Code Repository, Conducting the Project Postmortem, Releasing the Software Project Manager, Closing the Project,The Role of the Organization in Project Closure,The Project Management Office,The Configuration Control Board,The Systems Administration Department,Reference.
Agile Project Management:Introduction, Project Management Roles, Agile Project Management Characteristics, Metaphor,Teamwork and Collaboration, Guiding Principles, Open Information, Use a Light Touch, Monitoring and Adjustment, The Nuts and Bolts of Agile Project Management, Planning the Work, Controlling the Work, Process Improvement, Reference.

TUTORIALS:

Writing Cases for the following.

1. Writing requirement Proposal.
2. WritingNegotiationProposal.
3. Writing Feasibility Study Proposal.
4. Software Project Planning Proposal.
5. Software Project Execution Control Proposal.
6. WritingSchedules.
7. Software Project Closure

TEXT BOOKS / REFERENCES:

Text books:

1. **"**Mastering Software Project Management: Best Practices, Tools and Techniques**"**, MuraliChemuturi, Thomas M. Cagley, J. Ross Publishing, 2010,

Reference books:

1. "IT Project MAnagement – On track from Start to Finish", Book by Joseph Phillips, 2002.
2. "Managing the unmanageable" by Mantle and Lichty, 2012.
3. Making Things Happen: Mastering Project Managementby Scott Berkun, 2008.

ADDITIONAL LEARNING SOURCES:

1. https://www.tutorialspoint.com/software_engineering/software_project_management.htm
2. http://searchsoftwarequality.techtarget.com/tutorials/Software-Project-Management-Process

CO - PO MAPPING:

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

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	Laboratory	01	00	03	00	48

Course No	Course Title	Pre Requisites
MCA51L	.NET Technologies Laboratory	OOPS,Java,HTML

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: 2 Practical 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:

CO1: Develop application using the concept of .NET framework and basics of C# .NET.
CO2: Create server side applications using C#.NET.
CO3: Develop web applications using the ASP.NET.
CO4: Comprehend ASP.NET web form, state management and error handling mechanism.
CO5: Access and manipulate data in a database by using Microsoft ADO.NET

LIST OF PROGRAMS TO BE COVERED:

C# Programs:

1. Write a console application that obtains four int values from the user and displays the product.

2. If you have two integers stored in variables var1 and var2, what Boolean test can you perform to see if one or the other (but not both) is greater than 10?
3. Write an application obtains two numbers from the user, and displays them, but rejects any input where both numbers are greater than 10 and asks for two new numbers.
4. Write a console application that places double quotation marks around each word in a string.
5. Write an application that uses two command-line arguments to place values into a string and an integer variable, respectively. Then display these values
6. Write an application that receives the following information from a set of students:
   Student Id, Student Name, Course Name, Date of Birth:
   The application should also display the information of all the students once the data is Entered. Implement this using an Array of Structures.

7. Write programs using conditional statements and loops:

   a) Generate Fibonacci series
   b) To Check Prime Numbers
   c) Generate Prime Numbers till the given numbers
   d) Check the given number is palindrome
   e) Find sum of digits of a number.

8. Write a program to declare class 'Distance' have data members dist1,dist2 ,dist3. Initialize the two data members using constructor and store their addition in third data member using function and display addition.
9. Write a program using function overloading to swap two integer numbers and swap two float numbers
10. Write a program to implement single inheritance
11. Define a class 'salary' which will contain member variable Basic, TA, DA, HRA. Write a program using Constructor with default values for DA and HRA and calculate the salary of employee.
12. Program to implement the following multiple inheritance using interface
13. Write a program to implement multilevel inheritance
14. Write a program to create a delegate called TrafficDel and a class called TrafficSignal with the delegate methods to display appropriate operations for RED, YELLOW and GREEN signals.
15. Write a program to accept a number from the user and throw an exception if the number is not an even number
16. Create an application that allows the user to enter a number in the textbox named 'getnum'. Check whether the number in the textbox 'getnum' is palindrome or not. Print the message accordingly in the label control named lbldisplay when the user clicks on the button 'check'.
17. Create an application which will ask the user to input his name and a message, display the two items concatenated in a label, and change the format of the label using radio buttons and check boxes for selection , the user can make the label text bold ,underlined or italic and change its color . include buttons to display the message in the label, clear the text boxes and label and exit.
18. List of employees is available in listbox. Write an application to add selected or all records from

listbox (assume multi-line property of textbox is true).

ASP.NET Programs:

1. Write a ASP.NET code snippet for displaying "*Hello World!*".
2. Write a simple ASP.NET program to display the following Web Controls:
   a) A button with text "click me". The button control must be in the center of the form. b) A label with a text hello c) A checkbox. The form name must be Web Controls.
3. Write a program that displays a button in green color and it should change into yellow when the mouse moves over it.
4. Create a static web page of your choice and add different controls populated with data using ASP.NET

ASP.NET, C#.NET & ADO.NET Programs

1. Create a ASP.NET web form for User Login with fields, Usr_Name, Password and create appropriate messages for valid and invalid login. Use ASP.NET, C#.NET & ADO.NET with MySQL as DB.
2. Create a student registration system with following fields; Stud_ID, Stud_Name, Course_Name, Age, Place, Blood group, Fees_Paid. Use ASP.NET, C#.NET, ADO.NET and MySQL as database.
3. Create an Employee Salary generation system with fileds Emp_ID, Emp_Name, Emp_DOB, Emp_DOJ, Designation, Department, Place. Use ASP.NET, C#.NET and ADO.NET with MySQL as DB
4. Create an Inventory Management System with fields; Item_ID, Item_Name, Min_Qty, Max_Qty, ReOrd_Qty, Exp_Date, Price. Create a web application which should receive & issue items and display appropriate message for reaching minimum, maximum & reorder level quantity and also list out the expired items. Use ASP.NET, C#.NET, ADO.NET with MySQL as Database.

TEXT BOOKS / REFERENCES:

Text Books:

1. Matthew MacDonald. Beginning ASP.NET 4 in C# 2010, APRESS, 2010

Reference Books:

1. Joseph Mayo. Visual studio 2010 - A beginners guide – BPB Publications 2010
2. Greg Buczek: ASP.Net Developer's Guide, Tata McGraw Hill Edition 4th Edition, 2005.
3. Pro ASP.NET 4 in C# 2010, MacDonald and Freeman

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:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H	H	H	H	H	H	H	M	M	M	H	M
CO2	H	M	H	M	H	M	H	L	M	M	M	L
CO3	H	M	H	M	H	M	H	L	M	M	M	L
CO4	H	M	M	M	H	M	H	L	M	M	M	L
CO5	H	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	Laboratory	04	04	00	03	48

Course No	Course Title	Pre Requisites
MCA52L	Mobile Applications Laboratory	Java

COURSE ASSESSMENT METHOD:

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

COURSE OUTCOMES:

Upon successful completion of this course, students will be

CO1: Gain broad understanding of the discipline of Mobile Application Development using J2ME Technology.

CO2: Develop User Interface for a J2ME application

CO3: Manage data on both service-side components and client-side applications and Address Portability and Compatibility issues between PDA'S and Cell phones.

CO4: Implement the design using Android SDK.

CO5: Implement the design using Objective C and iOS

TOPICS COVERED:

List of Experiments on J2ME Platform:

1. Installation of Java Wireless Toolkit (J2ME).

2. Working with J2ME Features:
   - Menu boxes.
   - Event handlings.
   - Input checking.
   - Threads and high level UI.
   - High level UI.
3. Create a MIDP application, where the user can enter player name and points. The program saves the information to the record using RMS at MIDP device. Program should also print out the top 10 player list to the end user. You can use this class in your game if you made own class for saving and reading record sets.
4. Working on Drawing and Images.
5. Create a MIDP application, which draws a bar graph to the display. Data values can be given at int[] array.
6. Developing Networked Applications using the Wireless Toolkit
7. Authentication with a Web Server.
8. Login to HTTP Server from a J2ME Program.
9. Web Application using J2ME.

List of Programs on Android and IOS:

1. Develop an application that uses GUI components, Font and Colours
2. Develop an application that uses Layout Managers and event listeners.
3. Develop a native calculator application.
4. Write an application that draws basic graphical primitives on the screen.
5. Develop an application that makes use of database.
6. Develop an application that makes use of RSS Feed.
7. Implement an application that implements Multi threading
8. Develop a native application that uses GPS location information.
9. Implement an application that writes data to the SD card.
10. Implement an application that creates an alert upon receiving a message.
11. Write a mobile application that creates alarm clock

TEXT BOOKS / REFERENCES:

Text Books:

1. James Keogh , J2ME The Complete Reference , Tata McGrawHill.
2. Charlie Collins, Michael Galpin and Matthias Kappler, “Android in Practice”, DreamTech, 2012.
3. David Mark, Jack Nutting, Jeff LaMarche and Frederic Olsson, “Beginning iOS.

Reference Books:

1. Michael Juntao Yuan, Enterprise J2ME, Developing Mobile Java Applications Pearson Education , 2011.
2. Sing Li, Jonathan B. Knudsen, Beginning J2ME: From Novice to Professional, Third Edition, Apress, 2015.
3. Development: Exploring the iOS SDK”, Apress, 2013.

ADDITIONAL LEARNING SOURCES:
http://developer.android.com/develop/index.html.

CO-PO MAPPING:

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

JSS Science and Technology University, Mysuru
Department of Mater of Computer Applications [MCA]

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
MCA54L	Current Trends in IT Laboratory (Exploration and Presentation)	

COURSE ASSESSMENT METHOD:

1. Internal Assessment [5 Events: Exploration, Implementation and Presentation]

   Total Marks: 50 [10* 5].

COURSE OUTCOMES:

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

CO1: Exploration of different applications related to Cloud Computing, Big Data, Machine intelligence etc.
CO2: Conduct thorough survey on current trends with their futuristic needs.
CO3: Apply relevant architecture based on the application domain.
CO4: Design system by implementing agile standards
CO5: Present the findings and possible outcomes with benefits.

TOPICS COVERED:

Students are expected to build an application by using the latest trends topics in Information technology

- Machine Intelligence applications, Cloud Computing, Big Data or any new trends in IT.
- Different Programming Languages.

- Database like MySQL, Oracle, MS SQL etc.

ADDITIONAL LEARNING SOURCES:

1. http://www.managementstudyguide.com/emerging-trends-in-information-technology.html
2. http://www.itbusinessedge.com/slideshows/show.aspx?c=87336
3. http://www.hangthebankers.com/top-10-information-technology-trends-2016/
4. http://www.networkcomputing.com/storage/10-hot-technology-trends-2016/520323524
5. http://www.gocorptech.com/news-events/it-solutions-news-current-trends-of-information-technology/
6. http://www.csc.com/innovation/insights/135754-it_trends_to_watch_in_2016
7. http://www.forbes.com/sites/gartnergroup/2016/01/15/top-10-technology-trends-for-2016/#2c60155d5ae9

CO-PO MAPPING:

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

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.

2. 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.

3. 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.

4. 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.

5. 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.

6. 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?

  - 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.

  - 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.

- 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.

- Intended users: Who?

This should address who is the user of the system/beneficiary of the work.

- Environment: Where?

This should address area of domain, your application or work will be useful.

7. 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

3)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 (Guide (50%) + Panel (50%))	Report Evaluation		Report Evaluation	Viva Voce	Total Marks
		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	**2**
Report Evaluation.{Average of internal and external examiners}	**5**
Viva Voice	**2**
Total	**15**

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.