



**RAMAIAH**  
Institute of Technology

# **CURRICULUM**

**for the Academic year 2020 – 2021**

**MASTER OF COMPUTER APPLICATIONS**

**III & IV SEMESTER MCA**

**RAMAIAH INSTITUTE OF TECHNOLOGY**  
(Autonomous Institute, Affiliated to VTU)  
Bangalore – 560054.

## About the Institute

Dr. M. S. Ramaiah a philanthropist, founded 'Gokula Education Foundation' in 1962 with an objective of serving the society. M S Ramaiah Institute of Technology (MSRIT) was established under the aegis of this foundation in the same year, creating a landmark in technical education in India. MSRIT offers 13 UG programs and 15 PG programs. All these programs are approved by AICTE. All the UG programs & 09 PG programs are accredited by National Board of Accreditation (NBA). The institute is accredited with 'A' grade by NAAC in 2014. University Grants Commission (UGC) & Visvesvaraya Technological University (VTU) have conferred Autonomous Status to MSRIT for both UG and PG Programs till the year 2029. The institute is a participant to the Technical Education Quality Improvement Program (TEQIP), an initiative of the Government of India. The institute has 380 competent faculty out of which 60% are doctorates. Some of the distinguished features of MSRIT are: State of the art laboratories, individual computing facility to all faculty members, all research departments active with sponsored funded projects and more than 300 scholars pursuing Ph.D. To promote research culture, the institute has established Centre of Excellence for Imaging Technologies, Centre for Advanced Materials Technology & Schneider Centre of Excellence. **M S Ramaiah Institute of Technology has obtained "Scimago Institutions Rankings" All India Rank 65 & world ranking 578 for the year 2020.**

The Centre for Advanced Training and Continuing Education (CATCE), and Entrepreneurship Development Cell (EDC) have been set up on campus to incubate startups. **M S Ramaiah Institute of Technology secured All India Rank 8<sup>th</sup> for the year 2020 for Atal Ranking of Institutions on Innovation Achievements (ARIIA), an initiative of Ministry of Human Resource Development(MHRD), Govt. of India.** MSRIT has a strong Placement and Training department with a committed team, a good Mentoring/Proctorial system, a fully equipped Sports department, large air-conditioned library with good collection of book volumes and subscription to International and National Journals. The Digital Library subscribes to online e-journals from Elsevier Science Direct, IEEE, Taylor & Francis, Springer Link, etc. MSRIT is a member of DELNET, CMTI and VTU E-Library Consortium. MSRIT has a modern auditorium and several hi-tech conference halls with video conferencing facilities. It has excellent hostel facilities for boys and girls. MSRIT Alumni have distinguished themselves by occupying high positions in India and abroad and are in touch with the institute through an active Alumni Association.

**As per the National Institutional Ranking Framework, MHRD, Government of India, M S Ramaiah Institute of Technology has achieved 59<sup>th</sup> rank among 1071 top Engineering institutions of India for the year 2020 and 1<sup>st</sup> rank amongst Engineering colleges(VTU) in Karnataka.**

## **About the Department**

The Department of Master of Computer Applications was established in 1997 with the objective of producing high quality professionals to meet the demands of the emerging field of Computer Applications. The department got academic autonomy in the year 2007 and is accredited by NBA. The department is recognized as a Research Centre under Visvesvaraya Technological University in 2012.

### **VISION OF THE INSTITUTE**

To be an Institution of International Eminence, renowned for imparting quality technical education, cutting edge research and innovation to meet global socio economic needs

### **MISSION OF THE INSTITUTE**

MSRIT shall meet the global socio-economic needs through

1. Imparting quality technical education by nurturing a conducive learning environment through continuous improvement and customization
2. Establishing research clusters in emerging areas in collaboration with globally reputed organizations
3. Establishing innovative skills development, techno-entrepreneurial activities and consultancy for socio-economic needs

### **QUALITY POLICY**

We at M. S. Ramaiah Institute of Technology strive to deliver comprehensive, continually enhanced, global quality technical and management education through an established Quality Management System complemented by the synergistic interaction of the stake holders concerned

### **VISION OF THE DEPARTMENT**

To be a nationally prominent and internationally recognized department in academics and research activities with the aim of developing competitive software professionals to serve the society and ever changing industry.

### **MISSION OF THE DEPARTMENT**

1. To enable the students to be knowledgeable and creative through state-of-the-art curriculum and innovative teaching methodologies
2. To provide training programs that bridges the gap between academia and industry to produce competitive software professionals
3. To inculcate ethical values in the students enabling them to become socially committed professionals
4. To enhance the research quality and productivity, by providing required facilities

## **PROGRAM EDUCATIONAL OBJECTIVES (PEOs):**

**PEO1.** Excel in professional career in the field of Computer Applications and contribute to research and development activities.

**PEO2.** Provide software solutions that are socially acceptable and adapt emerging technologies and tools.

**PEO3.** Exhibit ethical and communication skills and engage in lifelong learning.

## **PROGRAM OUTCOMES (POs):**

**PO1:** Apply knowledge of mathematics and computing principles appropriately to develop conceptual model for real world problems.

**PO2:** Identify and formulate problem definition for real world problems, analyse the literature of the domain and provide solutions using mathematics and computing.

**PO3:** Design, develop and assess a software system, process, component, or program of varying complexity that meet specified needs with appropriate consideration for public health and safety and societal considerations.

**PO4:** Use research based knowledge to analyse and interpret data and synthesize information to provide valid conclusions.

**PO5:** Adapt to understand, select, use and create modern tools and technologies necessary for computing practices.

**PO6:** Adopt professional ethics, principles of professional computing practices, cyber regulations and responsibilities.

**PO7:** Recognize the need and engage in self-learning for continual development as a computing professional.

**PO8:** Apply the management principles for managing projects as an individual, as a member and as a leader in a team under multidisciplinary environments.

**PO9:** Communicate effectively about computing activities in both verbal and written form with the computing community and with society.

**PO10:** Assess the local and global impact of software solutions on individuals, organizations and society.

**PO11:** Perform effectively as an individual and as a member or leader in diverse teams and in multidisciplinary environments.

**PO12:** Adopt standardized software engineering practices with innovative ideas to succeed as an employee or entrepreneur.

### Curriculum Course Credits Distribution

Semester	Mathematics (MTH)	Management (MGT)	Humanities & Social Sciences (HSS)	Professional Courses		Project Work / Internship (PW /IN)	Seminar (SEM)	Self-Study (SS)	Total credits in a Semester
				Core (Hard core, Soft core, Lab) (PC- C)	Electives (PC-E)				
<b>First</b>	4		3	15				2	22
<b>Second</b>		3		18			1		22
<b>Third</b>				16	6				22
<b>Fourth</b>				12	9		1		22
<b>Fifth</b>				11	9				20
<b>Sixth</b>						22			24
<b>Total</b>	<b>4</b>	<b>3</b>	<b>3</b>	<b>72</b>	<b>24</b>	<b>22</b>	<b>2</b>	<b>2</b>	<b>132</b>

## SCHEME FOR 2019-2022 BATCH

### I SEMESTER

S. No	Course Code	Course Name	Credits*			
			L	T	P	Total
1.	MCA11	Problem Solving Using 'C'	2	1	1	4
2.	MCA12	Discrete Mathematics	3	1	0	4
3.	MCA13	Fundamentals of Computer Organization	4	0	0	4
4.	MCA14	Introduction to UNIX	3	0	1	4
5.	MCA15	Professional Communication and Ethics	2	1	0	3
6.	MCA16	Topics in Web Programming	0	1	2	3
<b>Total</b>			<b>14</b>	<b>4</b>	<b>4</b>	<b>22</b>

### II SEMESTER

S. No	Course Code	Course Name	Pre-requisite Course(s)	Credits*			
				L	T	P	Total
1.	MCA21	Object Oriented Programming with C++	MCA11	2	1	1	4
2.	MCA22 <sup>#</sup>	Data Analysis using Spreadsheets	-	0	1	2	3
3.	MCA23	Database Management Systems	-	3	0	1	4
4.	MCA24	Data Structures Using 'C'	MCA11	3	0	1	4
5.	MCA25	Management Information Systems	-	3	0	0	3
6.	MCA26 <sup>#</sup>	JavaScript Frameworks	MCA16	0	1	2	3
7.	MCAS1	Seminar I					1
<b>Total</b>				<b>11</b>	<b>3</b>	<b>7</b>	<b>22</b>

### III SEMESTER

S. No	Course Code	Course Name	Pre-requisite Course(s)	Credits*			
				L	T	P	Total
1	MCA31	Operating Systems		4	0	0	4
2	MCA32	Design and Analysis of Algorithms	MCA24	3	0	1	4
3	MCA33	Software Engineering		4	0	0	4
4	MCA34	Programming in Java	MCA21	3	0	1	4
5	MCAE--	Elective I					3
6	MCAE--	Elective II					3
<b>Total</b>							<b>22</b>

#### IV SEMESTER

S. No	Course Code	Course Name	Pre-requisite Course(s)	Credits*			
				L	T	P	Total
1	MCA41	Computer Networks	-	3	0	1	4
2	MCA42	Programming with Python	MCA16, MCA21	3	0	1	4
3	MCA43	Machine Learning	MCA12	3	0	1	4
4	MCAE--	Elective III					3
5	MCAE--	Elective IV					3
6	MCAE--	Elective V					3
7	MCAS2	Seminar II					1
<b>Total</b>							<b>22</b>

#### V SEMESTER

S. No	Course Code	Course Name	Pre-requisite Course(s)	Credits*			
				L	T	P	Total
1	MCA51	Object Oriented Modeling and Design Patterns	MCA21	3	0	1	4
2	MCA52	Information Security		3	0	1	4
3	MCASC-	Soft-core Course		3	0	0	3
4	MCAE--	Elective VI					3
5	MCAE--	Elective VII					3
6	MCAE--	Elective VIII					3
<b>Total</b>							<b>20</b>

#### SOFTCORE COURSES

S. No	Course Code	Course Name	Credits*			
			L	T	P	Total
1	MCASC1	Supply Chain Management	3	0	0	3
2	MCASC2	Software Project Management	3	0	0	3
3	MCASC3	Enterprise Resource Planning	3	0	0	3

#### VI SEMESTER

S. No	Course Code	Course Name	Credits*			
			L	T	P	Total
1.	MCA61	Project Work				22
<b>Total</b>						<b>22</b>

**Students have to acquire 2 credits by accomplishing any one of the following Co-Curricular Activities within the duration of the Programme (MCACCA):**

- On-line Certification Course
- Internship of 8-weeks followed by a report
- Paper Publication in International Conference along with a report and presentation
- Clearing one level in any of the recognized competitions



## ELECTIVE COURSES

S. No	Course Code	Course Name	Pre-requisite Course(s)	Credits*			
				L	T	P	Total
1.	MCAE01 <sup>#</sup>	User Interface and User Experience Design	MCA16	0	1	2	3
2.	MCAE02	Operations Research		2	1	0	3
3.	MCAE03 <sup>#</sup>	Hadoop Ecosystem		0	1	2	3
4.	MCAE04 <sup>#</sup>	Web Programming with PHP and AJAX	MCA16	0	1	2	3
5.	MCAE05 <sup>#</sup>	Computer Graphics and Animation		0	1	2	3
6.	MCAE06	Secure Coding in C and C++	MCA21	2	0	1	3
7.	MCAE07 <sup>#</sup>	NoSQL Databases	MCA23	0	1	2	3
8.	MCAE08 <sup>#</sup>	Mobile Application Development		0	1	2	3
9.	MCAE09 <sup>#</sup>	Web Component Development with J2EE	MCA34	0	1	2	3
10.	MCAE10	Programming with C#.Net	MCA21	2	0	1	3
11.	MCAE11	Digital Forensics		2	0	1	3
12.	MCAE12	System Modeling and Simulation		2	0	1	3
13.	MCAE13 <sup>#</sup>	Cloud Computing	MCA31, MCA41	0	1	2	3
14.	MCAE14 <sup>#</sup>	Software Testing	MCA33	0	1	2	3
15.	MCAE15 <sup>#</sup>	Programming IoT	MCA42	0	1	2	3
16.	MCAE16 <sup>#</sup>	ASP.Net with C#	MCA16, MCA21	0	1	2	3
17.	MCAE17	UNIX Systems Programming	MCA14, MCA31	2	0	1	3
18.	MCAE18	Deep Learning	MCA43	2	0	1	3
19.	MCAE19	Storage Area Networks		2	0	1	3
20.	MCAE20 <sup>#</sup>	Introduction to Big Data Analytics using SPARK		0	1	2	3
21.	MCAE21	Soft Skills and Personality Development		2	1	0	3
22.	MCAE22	Management and Entrepreneurship		3	0	0	3
23.	MCAE23	Agile Software Development	MCA33	2	0	1	3
24.	MCAE24	Social Network Analysis		2	0	1	3

**#Semester End Examination will be conducted for Laboratory**

**\* L: Lecture      T: Tutorial      P: Practical**

## **III SEMESTER**

# Operating Systems

**Course Code: MCA31**

**Pre requisite: Nil**

**Course Coordinator:**

**Credits: 4:0:0**

**Contact Hours: 56L**

## UNIT I

### **Introduction to Operating Systems, System Structure**

What operating systems do; Computer System Organization; Computer System Architecture; Operating System Operations; Computing Environments; Operating System Services; System Calls; Types of System Calls; System Programs; Operating System Structure; Virtual Machines; System boot.

### **Overview of Process**

Process Concept; Process Scheduling; Operations on Processes; Inter – Process Communication; Multi – Threaded Programming: Overview: Multithreading Models.

## UNIT II

### **Process Management**

Process Scheduling: Basic Concepts, Scheduling Criteria, Scheduling Algorithms, Multiple Processor Scheduling.

### **Process Synchronization**

Synchronization: The Critical Section Problem: Peterson's Solution; Semaphores; Classical Problems of Synchronization.

## UNIT III

### **Deadlocks**

Deadlocks: System model; Deadlock Characterization, Methods for handling deadlocks; Deadlock Prevention; Deadlock Avoidance; Deadlock Detection and Recovery from deadlock.

### **Memory Management**

Memory Management Strategies: Background, Swapping; Contiguous Memory Allocation; Paging; Segmentation; Virtual Memory Management; Background; Demand Paging; Page Replacement; Allocation of Frames; Thrashing.

## UNIT IV

### **File System, Implementation of File System**

File System: File concepts; Access methods, Directory Structure; File System Mounting; Protection; Implementing File System: File System Structure; File System implementation; Directory Implementation; Allocation Methods; Free Space Management.

## UNIT V

### **Secondary Storage Structures**

Mass Storage Structure, Disk Structures; Disk Attachment; Disk Scheduling; Disk Management; Swap Space Management.

### **Protection**

Protection: Goal of protection, Principles of Protection, Domain of Protection, Access Matrix, Implementation of Access matrix, Access Control, Revocation of Access Rights.

### **Text Books:**

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating Systems Principles, 8th Edition, Wiley – India.  
Chapters: 1.1, 1.2, 1.3, 1.5, 1.12, 2.1, 2.3, 2.4, 2.5, 2.7, 2.8.1, 2.8.2, 2.8.6, 2.11, 3.1 to 3.4, 4.1, 4.2, 5.1 to 5.3, 5.5.1, 5.5.2, 5.5.3, 6.1 to 6.3, 6.5, 6.6, 7.1 to 7.7, 8.1 to 8.4, 8.6, 9.1, 9.2, 9.4 to 9.6.1, 10.1 to 10.4, 10.6, 11.1 to 11.5, 12.1 to 12.6, 14.1 to 14.7

### **Reference Books:**

1. D M Dhamdhare: Operating Systems – A Concept Based Approach, 2nd Edition, Tata McGraw – Hill, 2002.
2. P C P Bhatt: Operating Systems, 2<sup>nd</sup> Edition, PHI, 2006.
3. Harvey M Deital: Operating Systems, 3<sup>rd</sup> Edition, Addison Wesley, 1990.

### **Course Outcomes (COs):**

1. Describe the elements and various functionalities of the operating system. (PO - 1,2,3)
2. Apply the techniques of process scheduling, process synchronization, and deadlocks. (PO - 1,2,3)
3. Demonstrate various memory allocation strategies and virtual memory techniques. (PO - 1,2,3)
4. Explain the physical and logical structure of the storage media, illustrate various algorithms for storage management. (PO - 1,2,3)
5. Outline the mechanisms of protection and security of various system resources. (PO - 1,2,3)

# Design and Analysis of Algorithms

**Course Code: MCA32**

**Prerequisite: MCA24**

**Course Coordinator:**

**Credits: 3:0:1**

**Contact Hours: 42L 28P**

## UNIT I

**Introduction:** Notion of Algorithm, Fundamentals of Algorithmic Problem Solving, Important Problem Types, Basics of data Structures.

**Fundamentals of the Analysis of Algorithm Efficiency:** Analysis Framework, Asymptotic Notations and Basic efficiency classes, Mathematical analysis of Recursive and Non recursive algorithms, Examples.

## UNIT II

**Brute Force:** Selection Sort, String Matching, Exhaustive Search.

**Divide-and-Conquer:** Merge sort, Quick sort, Binary Search, Multiplication of large integers.

**Principles of Parallel Algorithm Design:** Preliminaries-Decomposition, tasks, and Dependency graphs, Granularity, Concurrency and Task-Interaction, Decomposition Techniques.

## UNIT III

**Decrease-and-Conquer:** Insertion Sort, Depth First and Breadth First Search, Topological sorting, Algorithms for Generating Combinatorial Objects.

**Transform-and-Conquer:** Presorting, Heaps and Heap Sort.

## UNIT IV

**Space and Time Tradeoffs:** Sorting by Counting, Input Enhancement in String Matching, Hashing.

**Dynamic Programming:** Warshall's and Floyd's Algorithms.

**Greedy Technique:** Kruskal's Algorithm, Dijkstra's Algorithm, Huffman Trees.

## UNIT V

**Backtracking:** n-queens problem, Hamiltonian Circuit Problem, Subset-Sum Problem, General Remarks.

**Branch-and-Bound:** Knapsack problem, Traveling Salesman Problem.

**P, NP-Completeness and Approximation Algorithms:** Introduction, P and NP Problems, NP-complete problems, Approximation algorithms for the traveling salesman problem, Approximation algorithms for Knapsack problem.

## Laboratory

Programs that supplement the theory concepts are to be implemented.

**Text Books:**

1. Anany Levitin: Introduction to the Design and Analysis of Algorithms, 2nd Edition, Pearson Education, 2009.  
Chapters 1.1-1.4, 2.1-2.4, 3.1, 3.2, 3.4, 4.1-4.3, 4.5, 5.1-5.4, 6.1, 6.4, 7.1-7.3, 8.2, 9.1-9.4, 11.3, 12.1-12.3.
2. Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar: Introduction to Parallel Computing, 2nd Edition, Pearson Education, 2003.  
Chapters 3.1.1, 3.1.2, 3.2.

**Reference Books:**

1. Horowitz E., Sahani S., Rajasekharan S.: Fundamentals of Computer Algorithms, 2<sup>nd</sup> Edition, Universities Press, 2007.
2. Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein: Introduction to Algorithms, 3<sup>rd</sup> Edition, PHI, 2010.

**Course Outcomes (COs):**

1. Derive the time complexity of algorithms in terms of asymptotic notations. (PO - 1,2,4,5,9)
2. Apply the brute force, divide and conquer and parallel algorithm approaches for designing algorithm and determining the order of growth. (PO - 1,2,3,4,5,9)
3. Implement the decrease and conquer and transform and conquer approach for designing and determining the order of growth of algorithms. (PO - 1,2,3,4,5,9)
4. Demonstrate the design techniques dynamic programming and greedy technique to solve problems and determine the time complexity. Describe the space and time tradeoffs for algorithms. (PO - 1,2,3,4,5,9)
5. Apply the branch and bound and backtracking approach for solving problem and describe the concepts of NP-hard problem. (PO - 1,2,3,4,5,9)

# Software Engineering

**Course Code: MCA33**  
**Prerequisite: Nil**  
**Course Coordinator:**

**Credits: 4:0:0**  
**Contact Hours: 56L**

## UNIT I

**Introduction:** Professional software development, software engineering ethics, **Socio-technical systems:** Complex systems, System engineering, System procurement, System development, System operation, **Dependability and security:** Dependability properties, Availability and reliability, Safety, Security, **Software Process:** Software process model, Process activities, Coping with change, The rational unified process. **Product Life Cycle-** Product Life Cycle Stages, Product Life Cycle Case Study.

## UNIT II

**Requirements engineering:** Functional and Non-functional requirements, the software requirements document, Requirements specification, Requirements engineering processes, Requirements elicitation and analysis, Requirements validation, Requirements management.

**System modeling:** Context models, Interaction Models, Structural Models, Behavioral models, Model-driven engineering.

## UNIT III

**Software Design and Development, Architectural Design:** Architectural design decisions, Architectural views, Architectural patterns, Application architectures, **Design and implementation:** Object-Oriented design using the UML, Design patterns, implementation issues, Open source development, **Agile software development:** Agile methods, Plan-driven and agile development, Extreme programming, Agile project management, scaling agile methods.

## UNIT IV

**Software Evolution:** Evolution processes, Program evolution dynamics, Software maintenance, Legacy system management, **Software Reuse:** The reuse landscape, Application frameworks, Software product lines, COTS product reuse, **Software Testing:** Development testing, Test-driven development, Release testing, User testing.

## UNIT V

**Software Management, Project Management:** Risk Management, Managing people, Teamwork, **Project Planning:** Software pricing, Plan-driven development, Project scheduling, Agile planning, Estimation techniques, **Quality Management:** Software quality, Software standards, Software measurement and

metrics. **Configuration Management:** Change management, version management, system building, release management.

- *Assignment – Documentation covering phases of Software Development Life Cycle for real world problem.*
- *Internet resources or any relevant books can be used for Product Life Cycle.*

### **Text Books:**

1. Ian Sommerville: Software Engineering, 9<sup>th</sup> Edition, Pearson Education Publications, 2013.  
Chapters: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 16, 22, 23, 24,25

### **Reference Books:**

1. Roger. S. Pressman: Software Engineering-A Practitioners approach, 7<sup>th</sup> Edition, McGraw-Hill, 2010.
2. Shari Lawrence Pfleeger, Joanne M. Atlee: Software Engineering Theory and Practice, 4<sup>th</sup> Edition, Pearson Education, 2009.
3. Ali Behforooz, Frederick J. Hudson: Software Engineering Fundamentals, 1<sup>st</sup>Indian Edition, Oxford University Press, 2006.

### **Course Outcomes (COs):**

1. Exemplify the software engineering process, systems and security. (PO - 1,2,4,6,8,10,11)
2. Explain the software engineering process by identifying the requirements and system modeling. (PO - 1,2,3,4,6,8,10,11,12)
3. Design, develop and implement software based on requirements. (PO - 1,2,3,4,8,10,11,12)
4. Illustrate the evolution process, reuse and testing for giving a quality software to customer. (PO - 1,2,3,4,8,10,11,12)
5. Describe the software management, project management, and planning and configuration management. (PO - 1,3,4,8,11,12)



# Programming in Java

**Course Code: MCA34**  
**Prerequisite: MCA21**  
**Course Coordinator:**

**Credits: 3:0:1**  
**Contact Hours: 42L 28P**

## UNIT I

**Introducing Classes:** Introduction to Java, Class Fundamentals, Declaring Objects, Assigning Object Reference Variables, Introducing Methods, Constructors, The this Keyword, Garbage, The finalize() Method, Exploring the String Class, Using Command-Line Arguments, Varargs, Scanner class.

**Inheritance:** Inheritance Basics, Using super, Creating a Multilevel Hierarchy, When Constructors Are Called, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, Using final with Inheritance, The Object Class.

## UNIT II

**Packages and Interfaces:** Packages, Access Protection, An Access Example Importing Packages, Interfaces, Default Interface methods.

**Exception Handling:** Exception-Handling Fundamentals, Exception Types, Uncaught Exceptions Using try and catch, multiple catch Clauses, Nested try Statements, throw, throws, finally, Java's Built-in Exceptions, Creating Your Own Exception Subclasses.

## UNIT III

**Multithreaded Programming:** The Java Thread Model, The Main Thread, Creating a Thread, Creating Multiple Threads, Using isAlive() and join(), Thread Priorities, Synchronization, Inter-thread Communication, Suspending, Resuming, and Stopping Threads, Obtaining thread state, Using Multithreading

**Enumeration and Autoboxing:** Enumeration, Type Wrappers, Autoboxing.

**Generics:** What are Generics?, A Simple Generics Example, A Generics Class with two Type Parameters, The General Form of a Generic Class.

## UNIT IV

**The Collections Framework:** Collections Overview, The Collection Interfaces, The List Interface, The ArrayList Class, The LinkedList Class.

**Networking:** Networking Basics, Client server communication using TCP and UDP.

**Lambda Expressions:** Introducing Lambda Expressions, Block Lambda Expression.

## UNIT V

**Event Handling:** Two Event Handling Mechanisms, The Delegation Event Model, Event Classes, The event class, The Key event, Class Sources of Events,

Event Listener Interfaces, Using the Delegation Event Model, Adapter Classes, Inner Classes.

**JavaFX:** Introducing JavaFX GUI programming, Exploring JavaFX controls.

### **Laboratory**

Programs that supplement the theory concepts are to be implemented.

### **Text Books:**

1. Herbert Schildt: The Complete Reference JAVA, 9<sup>th</sup> Edition, TATA McGraw HILL, 2014.  
Chapters: 2, 6, 7, 8, 9, 10, 11, 12, 14, 15, 18, 22, 24, 34, 35

### **Reference Books:**

1. Paul Deitel and Harvey Deitel: Java How to Program, 9<sup>th</sup> Edition, PHI, 2012.
2. Y.Daniel Liang : Introduction to JAVA Programming, 6<sup>th</sup> Edition, Pearson Education, 2006
3. Cay S Horstmann, Gary Cornell: Core Java 2 volume 1 and volume 2, 7<sup>th</sup> Edition, Pearson Education, 2005.

### **Course Outcomes (COs):**

1. Develop programs using the OOP concepts and basic constructs in Java for a given problem. (PO-3,6)
2. Exemplify the usage of Packages, Interfaces and Exceptions for OO Programming. (PO-3, 6)
3. Apply Generics, Multithreading, Enumerations, and Auto boxing Concepts to develop robust programs. (PO-3, 6)
4. Implement the concepts of Networking, and analyze the usage of collection framework and lambda expression. (PO-3, 6)
5. Develop User Interfaces using JAVAFX with Event Handling for Dynamic Applications. (PO-3, 6)

# Operations Research

**Course Code: MCAE02**

**Prerequisite: Nil**

**Course Coordinator:**

**Credits: 2:1:0**

**Contact Hours: 28L 28T**

## UNIT I

**What is Operations Research?** Operations Research Models, Solving the OR models, Queuing & Simulation models, Art of Modelling, Phases of OR Study  
**Modelling with Linear Programming:** Two variable LP model, Graphical LP solution, Formulation of LP problems.

## UNIT II

**The Simplex Method:** LP model in equation form, Transition from graphical to algebraic solution, The Simplex method.

**Artificial starting solution - M Method, Two-Phase Method, Special cases in simplex method.**

## UNIT III

**Duality and Dual Simplex Method:** Definition of Dual Problem, Primal-Dual Relationships

**Game theory:** Introduction to Game Theory, the formulation of two persons, Zero sum games, solving simple Zero sum games, Games with mixed strategies.

## UNIT IV

**Transportation model and its Variants:** Definition of the Transportation Model, Mathematical formation of Transportation problem, The Transportation Algorithm: Determination of the starting solution, Iterative computations of the Transportation Algorithm, The assignment model - The Hungarian Method.

## UNIT V

**PERT and CPM:** Network representation, Critical Path (CPM) Computations, Construction of the Time Schedule, PERT calculations.

### Text Books:

1. Operations Research: An Introduction, Hamdy A Taha, 8<sup>th</sup> Edition, Pearson Education, 2011.  
Chapters: 1.1-1.6, 2.1, 2.2, 2.3.1, 2.3.3, 2.3.4 (Single-Period Production Model), 2.3.6, 3.1-3.5, 3.6.1, 4.1, 4.2, 4.3, 4.4.1, 5.1, 5.3.1, 5.3.2, 5.4.1, 6.1-6.3.2, 6.4.1, 6.4.2, 6.5.1-6.5.3, 6.5.5, 13.4 (Solution of Mixed Strategy Games using Graphical Method Only)

**Reference Books:**

1. S D Sharma: Operations Research, 15<sup>th</sup> Edition, Kedar Nath Ram Nath, Meerut, Delhi, 2005.
2. Fredrick S. Hiller, Gerald J Lieberman: Introduction to Operations Research, 9<sup>th</sup> Edition, McGraw Hill, India, 2008.

**Course Outcomes (COs):**

1. Formulate linear programming problems (LPP) and solve two-variable LPP graphically. (PO - 1,2,4)
2. Solve LP problems using simplex methods. (PO - 1,2,4)
3. Derive dual from a given primal problem and solve the LP problem using dual simplex method. Apply transportation and assignment model to obtain optimum solution. (PO - 1,2,4)
4. Construct network models for a given real world problem and determine the optimal solution. (PO - 1,2,4)
5. Apply PERT and CPM techniques for project management. Formulate and solve simple games to resolve competitive situations. (PO - 1,2,4)

# Web Programming with PHP and AJAX

**Course Code: MCAE04**

**Credits: 0:1:2**

**Prerequisite: MCA16**

**Contact Hours: 28T 56P**

**Course Coordinator:**

## Concepts to be covered in Tutorial

1. Overview of PHP, WAMP, LAMP, Syntactic characteristics, Primitives, Variables and Operators
2. Controlling program flow, Arrays, Strings, Dates and Times in PHP
3. Using functions and classes
4. Working with HTML forms and php
5. PHP and MySQL
6. Working with Cookies, Sessions and Headers
7. Overview of AJAX and its applications, Creating a simple AJAX example, XMLHttpRequest Object, Server side programming
8. Sending data to server using GET and POST
9. Demonstration of handling multiple XMLHttpRequest Objects
10. Demonstration of accessing XML data

## Reference Books:

1. VikramVaswani: PHP: A Beginner's Guide, Tata McGraw-Hill, 2017.
2. Steven Holzner: Ajax: A Beginner's Guide, Tata McGraw-Hill, 2017.
3. James Mallison: Mastering PHP 7, Packt Publishing Limited, 2017.
4. Antonio Lopez: Learning PHP 7, Packt Publishing Limited, 2016.

## Course outcomes (Cos):

- CO1. Implement various programming constructs of PHP (POs: 1,2,3,5,9)
- CO2. Demonstrate the concept of AJAX (POs: 1,2,3,5,9)
- CO3. Create web applications using PHP and AJAX (POs: 1,2,3,5,9,10,11,12)

# NoSQL Databases

**Course Code: MCAE07**

**Prerequisites: MCA23**

**Course Coordinator:**

**Credits: 0:1:2**

**Contact Hours: 28T 56P**

## Concepts to be covered in Tutorial

- Introduction to NoSQL
  - NoSQL Databases and their Benefits
- MongoDB Introduction
  - Introduction to MongoDB and its Features
  - Database, Collection and Documents
  - Data Types in MongoDB
  - Installation
  - The Mongo shell
- CRUD Operations
  - Create Operations
  - Read Operations
  - Update Operations
  - Delete Operations
  - Bulk Write
- Aggregation
  - Aggregation Pipeline
  - Map-Reduce
  - Single Purpose Aggregation Operations
- Text Search
  - Text Indexes
  - Text Search Operators
  - Text Search in the Aggregation Pipeline
- Geospatial Queries
  - Find Restaurants with Geospatial Queries
  - GeoJSON Objects
- Indexes
  - Single Field Indexes
  - Compound Indexes
  - Storage
- Replication
  - Replica Set Members
  - Replica Set Deployment Architectures
  - Replica Set High Availability
  - Replica Set Read and Write Semantics

- Sharding
  - Sharded Cluster Components
  - Sharding Strategy
  - Data Partitioning with Chunks
  - Sharded Cluster Balancer
- Web Application with MongoDB using PHP/Python/PyMongo
  - Connecting PHP/Python/PyMongo with MongoDB
  - CRUD Operations with PHP/Python//PyMongo
  - Aggregate functions with PHP/Python//PyMongo

### **Laboratory**

- Programs supplementing the concepts covered in tutorial.

### **References:**

1. Kristina Chodorow: MongoDB: The Definitive Guide: Powerful and Scalable Data Storage, 2<sup>nd</sup> Edition, O'Reilly, 2013.
2. <https://docs.mongodb.com/>
3. [https://www.tutorialspoint.com/mongodb/mongodb\\_tutorial.pdf](https://www.tutorialspoint.com/mongodb/mongodb_tutorial.pdf)

### **Course Outcomes (COs):**

1. Design and Develop queries for CRUD operations in MongoDB. (PO-1,2,3,4,5,7,11)
2. Design and Develop queries using Aggregation Framework and Pipeline to access MongoDB. (PO-1,2,3,4,5,7,11)
3. Develop PHP/Python//PyMongo programs for CRUD operations and Aggregation functions. (PO – 1,2,3,4,5,7,8,9,10,11,12)

# Digital Forensics

**Course Code: MCAE11**

**Prerequisite: Nil**

**Course Coordinator:**

**Credits: 2:0:1**

**Contact Hours: 28L 28P**

## UNIT I

**Understanding Cyber Crimes, Indian IT Act 2008, and its amendments, Computer Forensic and Investigations as a Profession:** Understanding Computer Forensics.

**Understanding Computer Investigations:** Preparing a Computer Investigation, Taking a Systematic Approach, Procedures for Corporate High-Tech Investigations, Understanding Data Recovery Workstations and Software.

## UNIT II

**Working with Windows and DOS Systems:** Understanding File Systems, Exploring Microsoft File Structures, Examining NTFS Disks, Understanding Whole Disk Encryption, Understanding the Windows Registry, Understanding Microsoft Startup Tasks, Understanding MS-DOS Startup Tasks, Understanding Virtual Machines.

**Macintosh and Linux Boot Processes and File Systems:** Understanding the Macintosh File Structure and Boot Process, Examining UNIX and Linux Disk Structures and Boot Processes, Understanding Other Disk Structures.

## UNIT III

**Current Computer Forensics Tools:** Evaluating Computer Forensic Tool Needs, Computer Forensics Software Tools, Computer Forensics Hardware Tools, Validating and Testing Forensics Software.

**Data Acquisition:** Understanding Storage Formats for Digital Evidence, Determining the best Acquisition Method, Contingency Planning for Image Acquisitions, Using Acquisition Tools, Validating Data Acquisitions, Performing RAID Data Acquisitions, Using Remote Network Acquisition Tools, and Using Other Forensic Acquisition Tools.

## UNIT IV

**Computer Forensics Analysis and Validation:** Determining What Data to Collect and Analyze, Validating Forensic Data, Addressing Data-Hiding Techniques, Performing Remote Acquisitions.

**Recovering Graphics Files:** Recognizing a Graphics File, Understanding Data Compression, Locating and Recovering Graphics Files, Identifying Unknown File Formats, Understanding Copyright Issues with Graphics.



## UNIT V

**Network Forensics:** Network Forensic Overview, Performing Live Acquisitions, Developing Standard Procedures for Network Forensics, Using Network Tools. **E-mail Investigations:** Exploring the Role of E-mail in Investigations, Exploring the Roles of the Client and Server in E-mail, Investigating E-mail Crimes and Violations, Understanding E-mail Servers, Using Specialized E-mail Forensics Tools.

*\*Assignment Based on Case Study*

### Laboratory

Lab exercises using forensic software and case study.

### Text Books:

1. Nelson, Phillips, Frank, Enfinger and Steuart: Computer Forensics and Investigations, Cengage Learning, 2008.  
Chapters: 1, 2, 4, 6, 7, 8, 9, 10, 11, 12

### Reference Books:

1. Marjie T. Britz: Computer Forensics and Cyber Crime - An Introduction, 2<sup>nd</sup> Edition, Pearson Education, 2012.
2. Pawan Duggal: Cyber Law- An exhaustive section wise Commentary on The Information Act along with Rules, Regulations, Policies, Notifications etc., Universal Law Publishing Co. Pvt. Ltd., 2014.
3. Harish Chander: Cyber Laws and IT Protection, PHI, 2012.

### Course Outcomes (COs):

1. Recall the Indian IT Act 2008, its amendments and describe various types of computer crime and its investigation techniques. (PO - 6,10)
2. Describe the file system and process handling concept of MS-DOS, Windows, and Macintosh and Linux operating system. (PO – 3,5,10)
3. Analyze and validate evidences using forensic tools. (PO – 4,5,6,9,10)
4. Extract, analyze hidden information from graphics, images and other files using forensic tools. (PO – 4,5,9,10)
5. Apply network forensic tools for network forensic, email investigation and live data forensic analysis. (PO – 4,5,9,10)

## **IV Semester**

# Computer Networks

**Course Code: MCA41**  
**Prerequisite: Nil**  
**Course Coordinator:**

**Credits: 3:0:1**  
**Contact Hours: 42L 28P**

## UNIT I

**Computer Networks and the Internet:** What Is the Internet? The Network Edge, The Network Core, Delay, Loss, and Throughput in Packet-Switched Networks, Protocol Layers and Their Service Models.

## UNIT II

**Application Layer:** Principles of Network Applications, The Web and HTTP, File Transfer: FTP, Electronic Mail in the Internet, DNS—The Internet's Directory Service, Peer-to-Peer Applications.

## UNIT III

**Transport Layer:** Introduction and Transport-Layer Services, Multiplexing and De-multiplexing, Connectionless Transport: UDP, Principles of Reliable Data Transfer, Connection-Oriented Transport: TCP.

## UNIT IV

**Transport Layer:** Principles of Congestion Control, TCP Congestion Control.  
**The Network Layer:** Introduction, Virtual Circuit and Datagram Networks, What's Inside a Router? The Internet Protocol (IP): Forwarding and Addressing in the Internet, Routing Algorithms.

## UNIT V

**The Network Layer:** Routing in the Internet Broadcast and Multicast Routing.  
**The Link Layer and Local Area Networks:** Link Layer: Introduction and Services, Error-Detection and Correction Techniques, Multiple Access Protocols, Link-Layer Addressing.

## Laboratory

Exercises to supplement the concepts using existing tools

### Text Books:

1. James F. Kurose, Keith W. Ross: Computer Networking: A Top-Down Approach, 5<sup>th</sup> Edition, Addison-Wesley, 2012.  
Chapters: 1, 2.1 - 2.6, 3, 4.1- 4.7, 5.1 - 5.4

### **Reference Books:**

1. Behrouz A. Forouzan: Data Communications and Networking, 4<sup>th</sup> Edition, Tata McGraw-Hill, 2006.
2. Alberto Leon-Garcia and Indra Widjaja: Communication Networks- Fundamental Concepts and Key architectures, 3rd Edition, Tata McGraw-Hill, 2004.
3. William Stallings: Data and Computer Communication, 8th Edition, Pearson Education, 2007.
4. Larry L. Peterson and Bruce S. David: Computer Networks – A Systems Approach, 4th Edition, Elsevier, 2007.
5. Wayne Tomasi: Introduction to Data Communications and Networking, Pearson Education, 2005.
6. Nader F. Mir: Computer and Communication Networks, Pearson Education, 2007.

### **Course Outcomes (COs):**

1. Describe basic terminologies used for computer networking. (PO - 1,2,3,5)
2. Demonstrate application layer protocols used for process to Process Communication and illustrate using packet tracer and wire shark. (PO - 1,2,3,5,9)
3. Illustrate transport layer protocols using wire shark. (PO - 1,2,3,5,9)
4. Elucidate network layer protocols using packet tracer. (PO - 1,2,3,5,9)
5. Explain link layer functionalities. (PO - 1,2,3,5)

# Programming with Python

**Course Code: MCA42**

**Prerequisites: MCA16, MCA21**

**Course Coordinator:**

**Credits: 3:0:1**

**Contact Hours: 42L 28P**

## UNIT I

### **Introduction to Python**

#### **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.

#### **Core Data Structures**

Strings, Lists, Tuples, and Dictionaries.

## UNIT II

### **Python Functions**

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 III

### **Object Oriented Programming**

Definition and defining a class, Constructor, Destructor, self and del keywords, Access to Attributes and Methods, getattr and setattr attributes, Data Attributes and Class Attributes, Data Hiding, Inheritance, Static Members.

#### **Regular Expressions**

Defining Regular Expressions and String Processing

## UNIT IV

### **File I/O**

File object attributes, Read and Write into the file, Rename and Delete a File.

#### **Exceptions Handling**

Handling Exceptions, Built-in Exceptions and User defined Exceptions.

#### **GUI Programming**

Introduction to Python GUI Programming, Tkinter Programming, Tkinter widgets, Events and Bindings.

## UNIT V

### **Working with Django**

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.

### **Laboratory**

Programs that supplement the theory concepts are to be implemented.

### **Reference Books:**

1. Timothy A. Budd: Exploring Python, Tata McGraw-Hill, 2011.
2. Ascher, Lutz: Learning Python, 4<sup>th</sup> Edition, O'Reilly, 2009.
3. Jeff Forcier, Paul Bissex, Wesley Chun: Python Web Development with Django, Addison-Wesley (e-book), 2008.
4. Wesley J Chun: Core Python Applications Programming, Pearson Education, 3rd Edition, 2013.

### **Course Outcomes (COs):**

1. Apply the basic and core data structures of Python programming. (PO – 1,2,3,5,9)
2. Develop modular programs using functions. (PO – 1,2,3,5,9)
3. Implement object oriented concepts and regular expressions in Python. (PO – 1,2,3,5,9)
4. Create graphical user interface using Tkinter and implement files and exception handling. (PO – 1,2,3,5,9)
5. Build web applications using Django framework. (PO – 1,2,3,5,9)

# Machine Learning

**Course Code: MCA43**  
**Prerequisites: MCA12**  
**Course Coordinator:**

**Credits: 3:0:1**  
**Contact Hours: 42L 28P**

## UNIT I

**Introduction:** Machine Learning, Types of Machine Learning, Applications of Machine Learning. Preparing to Model, Basic types of data in Machine Learning, Exploring structures of Data, Data quality and remediation, Data Pre-processing

## UNIT II

**Regression Model:** Correlation and Regression, Simple Linear Regression, Multiple Linear Regression, Overview of Logistic Regression, Model Validation using t-test

## UNIT III

**Classification:** Bayes Theorem, Naïve Bayes classifier, Building a Decision tree, Information gain of Decision tree, Algorithm for Decision Tree.

## UNIT IV

**Support Vector Machine:** Introduction to Support Vector Machines (SVM), Linear SVM, Optimal Hyperplane, Non-Linear SVM, Radial Basis Function

## UNIT V

**Clustering:** Partitioning Clustering Method, Hierarchical Clustering Method  
**Neural Networks:** Perceptron, Overview of Activation Functions, Multilayer Network.

## Laboratory

Implementation of techniques for Machine Learning

## Reference Books:

1. Saikat Dutt, Subramanian Chandramouli, Amit Kumar Das, “Machine Learning”, 2019.
2. Anuradha Srinivasaraghavan, Vincy Joseph,” Machine Learning”, Wiley, 2019.
3. Tom Mitchell, Machine Learning, 1st Edition, McGraw- Hill, 1997.
4. Ethem Alpaydin, Introduction to Machine Learning, 2nd Edition, The MIT Press Cambridge, Massachusetts London, England, 2010
5. Pang-Ning Tan, Michael Steinbach, and Vipin Kumar, “Introduction to Data Mining”, Second Edition, Pearson Education 2006.

6. Jiawei Han, Micheline Kamber, Jian Pei, “Data Mining: Concepts and Techniques”, 3rd Edition, Elsevier Publications, 2011.

**Course Outcomes (COs):**

1. Distinguish between Supervised, unsupervised and semi-supervised learning and prepare the data for analysis. (PO-1,2,4)
2. Develop Regression Models for a given problem. (PO-1,3,4,6)
3. Classify the future data trends using popular classification techniques. (PO1,3,4,6)
4. Analyse the usage of SVM in solving problems. (PO-1,3,4.6)
5. Group the data into clusters and design Neural Network Models. (PO-1,3,4,6)



# User Interface and User Experience Design

**Course Code: MCAE01**

**Prerequisite: MCA16**

**Course Coordinator:**

**Credits: 0:1:2**

**Contact Hours: 28T 56P**

## Topics to be Covered in Tutorial

- Usability of Interactive Systems
- Guidelines, Principles, and Theories
- Managing Design Processes
- Direct Manipulation and Virtual Environments
- Collaboration and Social Media Participation
- Balancing Function and Fashion
- UX- What is UX Designs? Why is UX so important? Full stack design, UX Design Process, Discovery and Planning, The UX STRATEGY, UX Research. UX analysis
- User Behavior Basics and User Research
- Designing Behavior.
- UI Design and Implementation

## Laboratory

Students have to design the User Interface components based on the concepts taught in tutorial and Design Website template and Navigation using: Adobe Photoshop and Adobe Image Ready, Adobe XD. Microsoft Visio

**CASE Tools:** Introduction to Adobe Photoshop, Adobe Image Ready, Adobe XD. Microsoft Visio, etc.

## List of Projects (Domain)

- Interface for online shopping website
- Mall Map
- e learning web site
- Tele-shopping
- Video/ Audio on demand web site
- Online banking
- ATM interface
- Automatic vending machine for Drinks
- Travel reservation system
- Booking of movie tickets
- Route finder
- Railway enquiry
- Students' Kiosk for institute's information
- Interface for waiter for billing (table wise etc.)
- Online buying of books.

- Online trading on Stock market
- Web site for buying Car
- University web site
- Week end holidays
- Pass port application tracking system
- Zoo information kiosk
- Museum Information Center
- Help desk for Hotel
- Hospital Management
- Servicing center for Automobile
- Patients information storage
- Website for Tuition class
- Catering Service (on-line chef)
- Marriage burro
- Placement agency
- Event management
- Web site for Device drivers and service center
- Just Dial type web site
- Website for promotions of new construction projects.

**Text Book:**

1. Shneiderman Plaisant Cohen Jacobs: Designing the User Interface, 5<sup>th</sup> Edition, Pearson Education, 2010.
2. Elvis Canziba - Hands-On UX Design for Developers: Design, prototype, and implement compelling user experiences from scratch, Packt Publishing, 2018.

**Course Outcomes (COs):**

1. Design the Usability of Interactive systems. (PO-3)
2. Develop Design process of Direct manipulation and Collaboration systems. (PO-3)
3. Design and Develop the UI of particular domain. (PO-3,6)

## **Hadoop Ecosystem**

**Course Code: MCAE03**

**Credits: 0:1:2**

**Prerequisite: Nil**

**Contact Hours: 28T 56P**

**Course Coordinator:**

### **Topics to be covered in Tutorial:**

- Hadoop Distributed File System and Hadoop Architecture
- Moving the Data into Hadoop and Moving the Data out from Hadoop
- MapReduce & HDFS
- Reading and Writing the files in HDFS using java program
- The Hadoop Java API for MapReduce: Mapper Class, Reducer Class and Driver Class
- Writing Basic MapReduce Program in java and Understanding the MapReduce Internal Components
- Hive Overview and Working with Hive
- Pig Overview and Working with Pig
- Moving the Data from RDBMS to Hadoop
- Moving the Data from RDBMS to HBase
- Moving the Data from RDBMS to Hive
- Flume Overview and Moving the Data from Web Server into Hadoop
- Real Time Example in Hadoop

### **Laboratory**

Programs supplement the tutorial concepts will be based on the latest version of Apache Hadoop

### **Reference Books:**

1. Tom White: Hadoop The Definitive Guide, Fourth Edition, 4<sup>th</sup> edition, O'Reilly Media, 2015.
2. Edward Capriolo, Dean Wampler, and Jason Rutherglen: Programming Hive, 1<sup>st</sup> edition, O'Reilly Media, 2012.
3. Alan Gates: Jason Ostrander: Programming Pig, 1<sup>st</sup> edition, O'Reilly Media, 2011.
4. Lars George: HBase The Definitive Guide, 1st edition, O'Reilly Media, 2011.

### **Course Outcomes (COs):**

1. Describe the Hadoop development framework and demonstrate the Hadoop shell commands. (PO-1,2,3,5)
2. Develop MapReduce programs in Java. (PO - 1,2,3,4,5)
3. Implement Hadoop Ecosystem to manage data (PO - 1,2,3,4,5,6,10)

# Secure Coding in C and C++

**Subject Code: MCAE06**

**Credits: 2:0:1**

**Prerequisite: MCA21**

**Contact Hours: 28L 28P**

**Course Coordinator:**

## UNIT I

**Running with Scissors**, Gauging the Threat, Security Concepts, Development Platforms, **Strings**, Character Strings, Common String Manipulation Errors, String Vulnerabilities and Exploits, Mitigation Strategies for Strings, String-Handling Functions, Runtime Protection Strategies, Notable Vulnerabilities.

## UNIT II

**Pointer Subterfuge**, Data Locations, Function Pointers, Object Pointers, Modifying the Instruction Pointer, Global Offset Table, The .ctors Section, Virtual Pointers, The atexit() and on\_exit() Functions, The longjmp() Function, Exception Handling, Mitigation Strategies, **Dynamic Memory Management**, C Memory Management, Common C Memory Management Errors, C++ Dynamic Memory Management, Common C++ Memory Management Errors.

## UNIT III

Memory Managers, Doug Lea's Memory Allocator, Double-Free Vulnerabilities Mitigation Strategies, Notable Vulnerabilities **Integer Security**, Introduction to Integer Security, Integer Data Types, Integer Conversions, Integer Operations.

## UNIT IV

Integer Vulnerabilities, Mitigation Strategies.

**Formatted Output**, Variadic Functions, Formatted Output Functions, Exploiting Formatted Output Functions, Stack Randomization, Mitigation Strategies, Notable Vulnerabilities.

## UNIT V

**File I/O**, File I/O Basics, File I/O Interfaces, Access Control, File Identification, Race Conditions, Mitigation Strategies, **Recommended Practices**, The Security Development Lifecycle, Security Training, Requirements Design, Implementation, Verification.

### Text Books:

1. Robert C. Seacord: Secure Coding in C and C++, 2<sup>nd</sup> Edition, Pearson, 2013. (Chapter 1,2,3,4,5,6,8,9)

### References:

1. SEI CERT Coding Standards

**Course Outcomes (COs):**

1. Identifying the vulnerability associated with Sting Handling and Mitigation strategies. (PO3)
2. Describe the threats associated with Runtime Memory Management and Pointers. (PO3)
3. Apply the mitigation techniques for vulnerable memory management and data type's conversion. (PO3)
4. Identify the vulnerability associated with Formatted Output and Concurrency. (PO3)
5. Describe the vulnerability mitigation techniques for File I/O and recommended practices for security development lifecycle. (PO3)

# Web Component Development with J2EE

**Course Code: MCAE09**

**Credits: 0:1:2**

**Prerequisites: MCA34**

**Contact Hours: 28T 56P**

**Course Coordinator:**

## Topics to be covered in Tutorial:

- JDBC and Embedded SQL
- Introduction to Servlets, Advantages of Servlets over CGI,
- The Servlets Life Cycle, Servlet API, Handling HTTP GET and POST Request.
- Servlet Context, Servlet Config, Request Dispatcher, Send Redirect
- Cookies, Session Tracking, Filter API
- Single Thread Model, Multi-tier Applications Using Database Connectivity
- Introduction to Java Server Pages(JSP), Advantages of JSP,
- Components of a JSP: Expressions, Scriptlets, Comments, Declaratives, Directives, (Page, Include, Taglib) Implicit Objects, JSTL
- JSP Standard Actions (usebean, setproperty, getproperty, param, plugin, and fallback)
- Introduction to Java Beans, The Java Beans API – Introspector, property Descriptor, Event Descriptor, Method Descriptor, A Bean Example, JSP with Java Beans
- Introduction to Springs, Spring MVC architecture, Configuring Spring with Eclipse, Crating first Spring Application
- Understanding annotations - @Controller, @RequestMapping, @PathVariable, @RequestParam, @ModelAttribute
- Introduction to Hibernate, Configuring Hibernate with Eclipse, creating first model using Hibernate, creating tables and records using Hibernate annotations.
- Sample application development using spring and hibernate.

## Laboratory

- Programs that supplement the tutorial concepts are to be implemented.
- Mini Project to be developed by the students.

## Text Books:

1. Jim Keogh: The complete Reference J2EE, 1<sup>st</sup>Edition, Tata McGraw Hill, 2002.
2. Mahesh P. Matha JSP and Servlets, 1<sup>st</sup>Edition, PHI, 2013.

3. Sharanam Shah Spring 3 with Hibernate 4 Project For Professionals .
4. Ranga Karanam Mastering Spring 5.0 Paperback – 28 Jun 2017.

### **Reference Books:**

1. Web Technologies Black Book, Kogent Learning solutions Inc sol., Dreamtech Press, 2012.
2. James McGovern, Rahim Adata, Yakov Fain, Jason Gordon, Ethan Henry, Walter Hurst, Ashish Jain, Mark Little, VaidyanathanNagarajan, Harshad Oak, Lee Anne Phillips: Java 2 Enterprise Edition 1.4 (J2EE 1.4) Bible, Wiley, 2003.
3. Bond, Law, Longshaw, Haywood, Roxburgh: Teach Yourself J2EE (J2EE 1.4), 2<sup>nd</sup> Edition, Pearson Education, 2005.

### **Course Outcomes (COs):**

1. Develop JDBC, Server side components using Java servlet with the interaction of different tiers. (PO- 1,2,3,5,7,9,10,11,12)
2. Apply the concepts JSP, Beans and Java frameworks for Dynamic Web Applications.  
(PO-1,2,3,5,7,9,10,11,12)
3. Create Simple web-based application for a real world problem using the concepts of servlet, Java Beans, JSP and frameworks with the interaction of multi-tier architecture. (PO-1,2,3,5,7,9,10,11,12)

# Programming with C#.Net

**Course Code: MCAE10**

**Prerequisite: MCA21**

**Course Coordinator:**

**Credits: 2:0:1**

**Contact Hours: 28L 28P**

## UNIT I

**Introduction to .NET Framework:** The.Net Strategy, The Origins of .Net technology, .Net Framework, The Common Language Runtime, Framework Base Classes, user and Program Interfaces, Visual Studio .Net, .Net Languages, Benefits of the .Net Approach.

**Overview of C#:** Introduction, a Simple C# Program, Namespaces, Adding Comments.

**Introducing Classes and Objects:** Class Fundamentals, Reference Variables and Assignment, Methods, Constructors, Garbage Collection and Destructors, this Keyword.

## UNIT II

**Managing Console I/O Operations:** Introduction, The Console Class, Console Input, Console Output, Formatted Output, Numeric Formatting, Standard Numeric format, Custom Numeric Format.

**Arrays and Strings:** Jagged Arrays, Assigning Array References, Using the Length Property, Implicitly Typed Arrays, For each Loop, Strings.

**A Closure Look at Methods and Classes:** Controlling Access to Class Members, Pass references to Methods, Use ref and out Parameters, use a variable Number of Arguments, The Main() Method, Understanding Static, Static Classes

## UNIT III

**Indexer and Properties:** Indexers, Properties.

**Inheritance:** Basics, Member Access and Inheritance, Constructors and Inheritance, Inheritance and Name hiding, Creating a Multilevel Hierarchy, Base Class References and Derived Objects, Virtual Methods and Overriding, Using Abstract Classes, Using Sealed to Prevent Inheritance, the Object Class.

**Interfaces, Structures and Enumerations:** Interfaces, Enumerations

## UNIT IV

**Namespaces:** Declaring a Namespace, Resolving name conflict using namespace, using, A Second Form of using, Nested Namespaces, Global Namespace, Using the::Namespace Alias Qualifier.

**Exception Handling:** The System Exception Class, Exception Handling Fundamentals, Consequences of an Uncaught Exception, Using Multiple Catch Clauses, Catching All Exceptions, Nested Try Blocks, Throwing an Exception, Using Finally.



## UNIT V

**LINQ:** LINQ Fundamentals, Filter Values with where, Sort Results with Order-by, Select and Nested From Clauses, Group Results With Group, Use into to Create a Continuation, Use let to Create a Variable in a Query, Join Two Sequences with Join, Anonymous Types, Create a Group Join.

**Delegates, Events and Lamda Expressions:** Delegates, Anonymous Functions, Anonymous Methods, Lambda Expressions, Events.

### Laboratory

Programs supplementing the theory concepts.

#### Text Books:

1. Herbert Schildt: The Complete Reference C# 4.0, Tata McGraw Hill, 2010. (Ch:1,6,7,8,10,11,12,13,15)
2. E. Balagurusamy: Programming in C#, Tata McGraw Hill, 4<sup>rd</sup> Edition, 2011. (Ch: 1,2,3)

#### Reference Books:

1. Joseph Albahari & Ben Albahari: C# 6.0 in a NutShell A Definitive Reference, O'Reilly, 6th Edition, 2016.
2. Karli Watson, Christian Nagel, Jacob Hammer Pedersen, Jon D. Reid, Morgan Skinner: Beginning Visual C#® 2010 Wrox publishers

#### Course Outcomes (COs):

1. Demonstrate the basic programming constructs of C# on .Net framework. (PO-1,2,3,5,9)
2. Apply object oriented concepts to develop dynamic interactive C# applications. (PO-1,2,3,5,9)
3. Demonstrate Indexers and Properties to manage access to class' instance data. (PO-1,2,3,5,9)
4. Apply C# exception handling model and using namespace to develop robust programs. (PO-1,2,3,5,9)
5. Implement delegate and events for method encapsulation and queries for processing in-memory data using LINQ. (PO-1,2,3,5,9)

# Software Testing

**Course Code: MCAE14**

**Prerequisite: MCA33**

**Course Coordinator:**

**Credits: 0:1:2**

**Contact Hours: 28T 56P**

## Concepts to be covered in Tutorial

- Articulate the basics of Software Testing Life Cycle using Manual Testing implementations.
- Installation of Selenium IDE, Recording and running test cases using Selenium IDE, Selenium Commands
- Installation of Selenium Webdriver in Pycharm and basics of python programming
- Finding Elements by NAME, ID, CSS, XPATH, LINKTEXT. Understanding “By” class
- Working with Web Elements; Useful Methods and Properties; Wait Types.
- Construct the complete automation framework in selenium webdriver.
- Running Complete Test suites.

## Laboratory

Students should demonstrate the working of manual testing, selenium IDE and should be able to test the given web page using selenium web driver and generate the necessary documents/ tables

**Project:** A team of 1 or 2 students must take up a Web Application and generate the necessary documents/tables using Manual Testing, Selenium IDE, Selenium Web Drivers and should run the complete test suite.

## Text Books:

1. Selenium with Python: A Beginners’ Guide, Pallavi R Sharma, BPB Publications.
2. David Burns: Selenium 2 Testing Tools: Beginner’s Guide, Packt Publishing, 2012.

## Course Outcomes (COs):

1. Explain the basics of the testing process and demonstrate the process of manual testing. (PO 1,2,3,4,5)
2. Demonstrate the basics of working with Selenium IDE. (PO 1,2,3,4,5)
3. Demonstrate the basics of working with Selenium Web Driver. (PO 1,2,3,4,5)

# UNIX Systems Programming

**Course Code: MCAE17**

**Prerequisite: MCA14, MCA31**

**Course Coordinator:**

**Credits: 2:0:1**

**Contact Hours: 28L 28P**

## UNIT I

### **Introduction**

**UNIX and ANSI Standards:** The ANSI C Standard, The POSIX Standards, POSIX feature test macros, Limits Checking at Compile Time and at Run Time. The POSIX APIs, API Common Characteristics.

### **UNIX Files**

File Types, The UNIX and POSIX File System, The UNIX and POSIX File Attributes, UNIX Kernel Support for Files, Relationship of C Stream Pointers and File Descriptors, Hard and Symbolic Links.

## UNIT II

### **UNIX File APIs**

General File APIs, File and Record Locking, Directory File APIs, Device File APIs, FIFO File APIs, Symbolic Link File APIs.

## UNIT III

### **UNIX Processes**

The Environment of a UNIX Process: Introduction, main function, Process Termination, Command-Line Arguments, Environment List, Memory Layout of a C Program, Memory Allocation, Environment Variables, setjmp and longjmp Functions, getrlimit, setrlimit Functions.

### **Process Control**

Introduction, Process Identifiers, fork, vfork, Zombie Process, wait, waitpid, Race Conditions, exec Functions.

## UNIT IV

**Process Relationships:** Introduction, Terminal Logins, Network Logins, Process Groups, Sessions, Controlling Terminal, tcgetpgrp, tcsetpgrp, and tcgetsid Functions, Job Control.

### **Signals**

Introduction, The UNIX Kernel Support for Signals, signal (), signal sets, Signal Mask, sigaction, The sigsetjmp and siglongjmp Functions, Kill, Alarm.

## UNIT V

### **Daemon Processes**

Introduction, Daemon Characteristics, daemon Coding Rules, Error Logging,

### **Interprocess Communication**

Introduction, Pipes, popen, pclose Functions; Coprocesses; FIFOs; Stream Pipes, Client-Server Connection Functions.

## Laboratory

Programs that supplement the theory concepts are to be implemented.

### Text Books:

1. Terrence Chan: UNIX System Programming Using C++, PHI, 1999.  
Chapters 1, 5, 6, 7, 8, 9
2. W. Richard Stevens, Stephen A. Rago: Advanced Programming in the UNIX Environment, 2nd Edition, Pearson Education, 2005.  
Chapters 7, 8, 9, 13, 15

### Reference Books:

1. Marc J. Rochkind: Advanced UNIX Programming, 2nd Edition, Pearson Education, 2005.
2. Maurice .J. Bach: The Design of the UNIX Operating System, Pearson Education, 1987.
3. Uresh Vahalia: UNIX Internals: The New Frontiers, Pearson Education, 2001.

### Course Outcomes (COs):

1. Describe UNIX file system, UNIX/POSIX standards and POSIX API's.  
(PO -1,2,4,5)
2. Implement API's required to perform various file operations on different file types. (PO - 1,2,3,5,9)
3. Implement API's required to create, run and control processes.  
(PO - 1,2,3,5,9)
4. Describe API's required for process relationships and signals.  
(PO - 1,2,3,5,9)
5. Describe various methods for handling daemon process and inter-process Communication. (PO - 1,2,4,5)

## Co-Curricular Activities

**Course Code: MCACCA**

**Credits: 2**

### **Guidelines:**

- Students have to acquire 2 credits by accomplishing any one of the following as a self-study component within the duration of the Programme:
  - On-line Certification Course
  - Internship of 8-weeks followed by a report
  - Paper Publication in International Conference along with a report and presentation
  - Clearing one level in any of the recognized competitions
- On-line Certification Course
  - Students can register for any On-line Certification Course in the field of Computer Science/Management for a minimum duration of 8 weeks
  - They have to submit the Course Completion Certificate along with their scores
- Internship of 8-weeks followed by a report
  - Students have to undergo the Internship in any Institute of National repute or any reputed/well-known industry
  - They are expected to submit a report and give a presentation
- Paper Publication in International Conference along with a report and presentation
  - Students have to publish a research paper in a peer-reviewed International Conference.
  - After the conference, he/she has to submit a report and give a presentation
- Clearing one level in any of the recognized competitions
  - Students have to register for any recognized competitions and clear one level
  - The committee has to identify suitable competition and make the students to aware of that.

### **Approval and Evaluation Process:**

- Approval and evaluation can be done by the committee along with the concerned proctor
- The students have to get prior approval from the committee to take up the activity
- A stipulated period can be provided for the approval and evaluation process