



**JECRC<sup>TM</sup>**  
**UNIVERSITY**  
BUILD YOUR WORLD

**Faculty of IT & Computer Application**

**Scheme & Syllabi**

**Master of Computer Applications ( MCA)**

**Specialization in Cyber Security**

**(In Association with EC Council)**

**Academic Programme**

**July 2020-22**

## **MCA**

**Total Credits for the Batch 2020-22 = 112 Credits**

- 1. Minimum Credits required = 102 Credits**
- 2. Total Relaxation = 10 Credits**
- 3. No relaxation in courses of type Core and Foundation and Skill Enhance Course subjects.**
- 4. Theory exams duration will be of 03 hours and Practical exams will be of 02 hours.**
- 5. Internal Assessment will be of 50 marks and End Term Assessment will be of 50 marks for theory and lab courses.**
- 6. The students will earn three Global Certifications of EC- Council for the specialization subjects mentioned in below scheme.**

<b>Semester wise Credit</b>				<b>Total Credits</b>	<b>Minimum Credits Required for Degree</b>
<b>III</b>	<b>IV</b>	<b>V</b>	<b>VI</b>		
<b>28</b>	<b>28</b>	<b>28</b>	<b>28</b>	<b>112</b>	<b>102</b>

## Semester – I

Course Code	Course Name	L (Hr.)	T (Hr.)	P (Hr.)	Credits	Type
MCA144B	Programming in Python	3	0	0	3	S
MCA119A	Programming in C++	3	0	0	3	CORE
MCA120B	Computer Networks	3	1	0	4	CORE
MCA121B	Advance Data Structures and Algorithms	3	0	0	3	CORE
MCA123B	Business Communication Skills	2	0	0	2	F
MCA147B	Programming in Python Lab	0	0	2	2	S
MCA125A	Programming in C++ Lab	0	0	2	2	S
MCA126A	Advance Data Structures and Algorithms Lab	0	0	2	2	S
MCA127B	Business Communication Skills Lab	0	0	2	2	S
<b>MCACS101</b> (Cyber Security Specialization )	<b>Certified Network Defender (CND)</b>	<b>3</b>	<b>0</b>	<b>2</b>	<b>5</b>	<b>CORE</b>
	<b>Total</b>	<b>17</b>	<b>1</b>	<b>10</b>	<b>28</b>	

❖ **F- Foundation, ID- Interdisciplinary, S- Specialization (Skill Enhance Courses)**

❖ **L- Lecture, T- Tutorial, P- Practical**

## Semester – II

Course Code	Course Name	L (Hr.)	T (Hr.)	P (Hr.)	Credits	Type
MCA182A	Soft Skills	3	0	0	3	F
MCA130A	Advance Java	3	0	0	3	CORE
MCA132B	PHP & MYSQL	3	0	0	3	CORE
MCA118A	Advance Database Management Systems	3	0	0	3	CORE
	Elective – II	3	1	0	4	S
MCA133A	Advance Java Lab	0	0	2	2	S
MCA134B	PHP & MYSQL Lab	0	0	2	2	S
MCA136A	Minor Project	0	0	2	2	S
<b>MCACS102 (Cyber Security Specialization )</b>	<b>Certified Ethical Hacker (CEH)</b>	<b>4</b>	<b>0</b>	<b>2</b>	<b>6</b>	<b>CORE</b>
	<b>Total</b>	<b>18</b>	<b>2</b>	<b>8</b>	<b>28</b>	

❖ **F- Foundation, ID- Interdisciplinary, S- Specialization (Skill Enhance Courses)**

❖ **L- Lecture, T- Tutorial, P- Practical**

## Semester – III

Course Code	Course Name	L (Hr.)	T (Hr.)	P (Hr.)	Credits	Type
MCA129A	Accounting Principles and Practices	3	0	0	3	ID
MCA183A	Agile Software Development	3	0	0	3	CORE
MCA138A	Information Security & Cyber Law	3	0	0	3	CORE
MCA184A	Data Science & Analytics	2	1	0	3	S
	Elective – IV	3	1	0	4	S
MCA185A	Data Science & Analytics Lab	0	0	2	2	S
MCA143A	Minor Project (PHP, ASP.Net, Java)	0	0	4	4	S
<b>MCACS103 (Cyber Security Specialization)</b>	<b>Certified Application Security Engineer (CASE) JAVA / .NET</b>	<b>4</b>	<b>0</b>	<b>2</b>	<b>6</b>	<b>CORE</b>
	<b>Total</b>	<b>18</b>	<b>2</b>	<b>8</b>	<b>28</b>	

❖ **F- Foundation, ID- Interdisciplinary, S- Specialization (Skill Enhance Courses)**

❖ **L- Lecture, T- Tutorial, P- Practical**

## Semester – IV

Course Code	Course Name	Credits	Type
MCA175A	Internship/Industrial Training/Project Presentation	28	CORE

## List of Courses for Elective – II & IV

Course Code	Course Name
MCA150A	Parallel Computing
MCA151A	Advance Computer Architecture
MCA152A	Theory of Computation
MCA153A	Data warehousing and Data Mining
MCA154A	Optimization Techniques
MCA155A	NLP and Information Retrieval
MCA156A	Object Oriented Analysis & Design
MCA157B	Cloud Computing
MCA158A	Distributed Computing
MCA159A	Introduction to Sales Force
MCA166A	Design & Analysis of Algorithms
MCA167A	Mobile Computing
MCA168A	Image Processing
MCA169A	Soft Computing
MCA170A	Compiler Design
MCA171A	Big Data Analytics
MCA172A	Organization Behavior
MCA173A	Artificial Intelligence and Expert System
MCA174A	Computer Graphics
SMCA178A	Animations
SMCA179A	Computer Network and Internet
SMCA180A	Cyber Security
MCA181A	Entrepreneurship
MCA186A	Robotics Process Automation

**Note:**

1. Students have to select electives as per the given list for Elective – I & III and Elective – II & IV.
2. Minimum Batch size is 20 for any Elective

## **Semester I**

**Course Name: Programming in Python**

**Course Code: MCA144B**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	0	0	3

### **Course Objectives:**

1. To outline the basics of python programming, Features, history, data types and variables.
2. To able to solve real-world problems through python programming.
3. To be able to exploit problem solving approaches, programming languages, object oriented programming.
4. To constructs of Python language such as control statements, functions, strings, files, data structures.
5. To apply the concept of Classes and objects, functions and array in python.

### **Syllabus**

#### **Unit I**

**Introduction to Procedural Programming:** Data types in Python: Comments in Python, identifiers, keywords, Integral Types, Integers, and Booleans Floating-Point Types: Floating-Point Numbers, Complex Numbers, Decimal Numbers, Strings, Comparing Strings, Slicing and Striding Strings, String Operators and Methods, Operators in Python, Input and Output

#### **Unit II**

**Collection Data Types:** Sequence Types, Tuples Named Tuples, Lists Set Types: Sets. Frozen Sets, Mapping Types: Dictionaries, Default Dictionaries, Ordered Dictionaries, Iterating and Copying Collections, Arrays in Python

### **Unit III**

**Control Structures and Functions** :Control Structures .Conditional Branching, Looping, Exception Handling, Catching and Raising Exceptions Custom Exceptions Custom Functions Names and Docstrings Argument and Parameter Unpacking, Accessing Variables in the Global Scope, Lambda Functions.

### **Unit IV**

**File handling:** Writing and Reading Binary Data, Pickles with Optional Compression, Raw Binary Data with Optional Compression, Writing and Parsing Text Files, Writing Text, Parsing Text, Parsing Text Using Regular Expressions, Writing and Parsing XML Files, Element Trees DOM (Document Object Model), Manually Writing XML Parsing XML with SAX (Simple API for XML), Random Access Binary Files ,A Generic BinaryRecordFile Class.

### **Unit V**

**Object Oriented Programming:** The Object-Oriented Approach, Object oriented approach Custom Classes, Attributes and Methods, Inheritance and Polymorphism, Using Properties to Control Attribute Access, Creating Complete Fully Integrated Data Types, Custom Collection Classes, Creating Classes That Aggregate Collections, Creating Collection Classes Using Aggregation, creating Collection Classes Using Inheritance, Python's Database Connectivity, Libraries in python for Machine Learning: Panda, Numpy, Matplotlib, Scikit-learn, Tensorflow.

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

**Upon successful completion of this subject students should be able to:**

CO1: Outline the basics of python programming, Features, history, data types and variables.

CO2: Understanding of Python especially Decision control, function and modules.

CO3: Understanding of the file handling and Data Structure.

CO4: Be exposed to object oriented concepts in Python programming, decision controls and function.

CO5: Understand the concept of class and object, class methods and Garbage.



**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF  
PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Core Python Programming, Black Book- Dreamtech, Dr. R. Nageswara Rao, 2017
2. Python Programming, Oxford, ReemaThareja, June 2017

**Reference Books**

1. “Python Testing Cookbook” by Greg L Turnquist
2. “Head First Programming” by Paul Barry and David Griffiths
3. “Python Crash Course: A Hands-On, Project-Based Introduction to Programming” by Eric Matthes.

**Course Name: Programming in C++**

**Course Code: MCA119A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	0	0	3

### **Course Objectives**

1. To explain the difference between object oriented programming and procedural programming and features of object oriented programming.
2. To be able to understand the program using more advanced C++ features such as composition of objects, operator overloads, dynamic memory allocation, inheritance and polymorphism, file I/O, exception handling, etc.
3. To be able to build C++ classes using appropriate encapsulation, objects and functions.
4. To be able to apply operator overloading in different form.
5. To understand the working of files, templates and exception handling.

### **Syllabus**

#### **Unit I**

**Introduction to Object Oriented Programming:** Generation of Programming, Programming Paradigms, Features of Object Oriented Programming , Introduction to C++, History of C++, Structure, First Program, Compiling and Executing C++, Using Comments, Tokens, Character Set, Keywords, Identifiers, Data Types, Variables, Constants, Enumerators, Input and Output Statements, Operators in C++, Operator Precedence and Associability, Decision Control and Looping Statements.

#### **Unit II**

**Classes, Objects and Functions:** Introduction , Class, Creating Objects, Accessing Object Members, Nested Member Functions, Making a Member Function Inline, Memory Allocation for Class and Objects, Array of Objects, DMA, Objects as Function Arguments, Returning Objects, this pointer, Constant Parameters and Members, Pointers Within a Class , Empty

Classes, Friend Classes, Constructor, Types of Constructors, Constructor with Arguments, Constructor Overloading, Destructors

### **Unit III**

**Operator Overloading and Type Conversions:** Scope of Operator Overloading, Syntax, Not Overloading Operators, Implementing Operator, Overloading Unary Operators, Overloading Binary Operators, Overloading Special Operators, Type Conversions,

### **Unit IV**

**Inheritance and Run-Time Polymorphism:** Defining Derived Classes, Access Specifiers, Inheritance, Types of Inheritance, Single Inheritance, Constructors and Destructors in Derived Class, Constructor in Multi-Level, Multi-Level Inheritance, Constructor in Multi-Level Inheritance, Multiple Inheritance, Hierarchical Inheritance, Multi-path Inheritance, Up-casting, Down-casting, and Cross-Casting, Run-time Polymorphism, Virtual Functions, Abstract Base Classes.

### **Unit V**

**File Handling, Templates & Exception Handling:** Streams in C++, Classes for File Stream, Opening and Closing of Files, Detecting the End-of-File, File Modes, File pointer, Use of Templates, Function Templates, Class Template, Class Templates and Friend Function, Templates and Static Variables in C++, Exception Handling, Multiple Catch Statements, Catch all Exceptions, Exceptions in Invoked Function

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

**Upon successful completion of this subject students should be able to:**

CO1: Understand the features of C++ supporting object oriented programming

CO2: Be able to program using more advanced C++ features such as composition of objects, operator overloads, dynamic memory allocation, inheritance and polymorphism, file I/O, exception handling, etc.

CO3: Be able to apply operator overloading in different form.

CO4: Understand how to apply the major object-oriented concepts to implement object oriented programs in C++, encapsulation, inheritance and polymorphism

CO5: Able to understand the working of files.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Let Us C: BalaGuruswamy, TATA McGraw Hill.
2. Object Oriented Programming with C++, Reema Thareja, Oxford University, August 2015.

**Reference Books**

1. Object Oriented Programming with C++, Sourav sahay, Oxford University, Sept 2012.
2. Robert Lafore, "Object-Oriented Programming in C++", Sams, Fourth Edition 2007
3. BjarneStroustrup, "The C++ Programming Language: Special Edition", Addison-Wesley, Third Edition 2000

**Course Name: Computer Network**

**Course Code: MCA120B**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

### **Course Objectives**

1. To be familiar with the terminology and concepts of Layering, Distributed Systems and Networks, Peer-to-Peer and Client-Server Networks.
2. To enhance the practical knowledge of protocols used in different layers.
3. To be familiar with physical layer based on telephone lines.
4. To be able to understand Error Detection, Error Correction, Flow Control.
5. To be able to understand the concept of Connection Oriented, Connectionless and routing algorithms.

### **Syllabus**

#### **Unit I**

**Introduction to Computer Networks:** Definition: Network , The Need of Resources Sharing, data communications components, data representation, and data flow, network topologies, categories of networks, organizations that set standards in data communications and networking, introduction to Open Systems Interconnection (OSI) and the Internet model (TCP/IP) layers and services.

#### **Unit II**

**The Physical Layer & Media:** Functions of physical layer, issues related to the physical layer and the transmission medium that is controlled by the physical layer, Bandwidth Utilization: Multiplexing and Spreading, Transmission Media, Switching.

#### **Unit III**

**The Data Link Layer:** Error Detection and Correction, Data Link Control, Multiple Access, Wired LANs: Ethernet, Wireless LANs.

## **Unit IV**

**The Network Layer:** Logical addressing: IPv4 and IPv6, Internet Protocol: IPv4, Datagram, Fragmentation, Checksum, IPv6 Advantages and Packet Format, Address mapping: ARP, RARp and DHCP, Error Reporting ICMP: Types of messages, message formats, ICMPv6 Error Reporting. Delivery, Forwarding, and Unicast and Multicast Routing.

## **Unit V**

**Transport Layer and Application Layer:** Process-to-Process Delivery: UDP, TCP, and SCTP. Congestion Control and Quality of service. Application Layer: Domain Name System, Remote Logging, Electronic Mail, and File Transfer, WWW and HTTP. Network Management: SNMP

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

**Upon successful completion of this subject students should be able to:**

CO1: To be familiar with the terminology and concepts of Layering, Distributed Systems and Networks, Peer-to-Peer and Client-Server Networks.

CO2: Describe, analyze and compare Physical Layer based on telephone lines.

CO3: Describe, analyze and compare a number of data link, network, and transport layer protocols, Error Detection, Error Correction and Flow Control.

CO4: Able to understand the concept of Connection Oriented, Connectionless and routing algorithms.

CO5: Enhance the practical knowledge of protocols used in different layers.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Behrouz A. Forouzan, Data Communication and Networking, Fifth Edition, Mcgraw Hill, 2017.
2. Andrew S. Tanenbaum, David J. Wetherall, Computer Networks, 5th Edition, Pearson, 2011.

**Reference Books**

1. Larry L. Peterson and Bruce S. Davie, Computer Networks: A System Approach, Fifth Edition, The Morgan Kaufmann Series in Networking, 2011
2. James Kurose and Keith Ross, Computer networking: A Top Down Approach, Seventh Edition, Pearson, 2017.
3. William Stallings, Data and Computer Communications, 10th Edition, Pearson, 2014

**Course Name: Advance Data Structure and Algorithms**

**Course Code: MCA121B**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	0	0	3

**Course Objectives**

1. To describe and implement a variety of advanced data structures (hash tables, priority queues, balanced search trees, graphs).
2. To demonstrate an understanding an understanding of tree data structure.
3. To demonstrate an understanding of advance tree and heap data structures
4. To understanding about graph data structure and its basic operations.
5. To understanding about Graph Applications.

**Syllabus**

**Unit I**

The Role of Algorithms in Computing, Analyzing algorithms, Designing algorithms, Growth of Functions, Asymptotic notations, DIVIDE AND CONQUER METHOD: Binary Search, Merge Sort, Quick sort and strassen's matrix multiplication algorithms. substitution method for solving recurrences, recursion-tree method for solving recurrences, master method for solving recurrences. Heap sort, Quick sort, Counting sort ,Radix sort , Bucket sort.

**Unit II**

Elementary Data Structures- Stacks and queues ,Linked lists, Hash Tables, Binary Search Trees- Insertion and deletion,B-trees- Definition of B-trees , Basic operations on B-trees, Deleting a key from a B-tree,Red-Black Trees- Properties of red-black trees, Rotations, Insertion, Deletion,

**Unit III**

**Dynamic Programming** - Matrix-chain multiplication , Elements of dynamic programming , Longest common subsequence , Optimal binary search trees, **Greedy Algorithms** - Elements of the greedy strategy ,Huffman codes, A task-scheduling problem.



## Unit IV

**Graph Algorithms** - Representations of graphs, Breadth-first search, Depth-first search, topological sort, Strongly connected components, Minimum Spanning Trees, Growing a minimum spanning tree, The algorithms of Kruskal and Prim, **Single-Source Shortest Paths** - The Bellman-Ford algorithm, Single-source shortest paths in directed acyclic graphs, Dijkstra's algorithm, Difference constraints and shortest paths, **All-Pairs Shortest Paths**- The Floyd-Warshall algorithm, **Maximum Flow**- Flow networks, The Ford-Fulkerson method

## Unit V

**String Matching** -The naive string-matching algorithm, The Rabin-Karp algorithm, The Knuth-Morris-Pratt algorithm, **NP-Completeness**-Polynomial time, Polynomial-time verification, NP-completeness and reducibility, NP-completeness proofs, NP-complete problems. **Approximation Algorithms** The vertex-cover problem, The traveling-salesman problem, the set-covering problem, Randomization and linear programming, The subset-sum problem.

## Course Outcomes (Cos):

**Upon successful completion of this subject students should be able to:**

CO1: Describe and implement a linear and non-linear data structures with its operations and sorting methods.

CO2: Demonstrate an understanding of tree data structure and Generate source code for binary search tree and heap.

CO3: To understand the concept of dynamic programming.

CO4: To understand the basic terminology of graph, graph traversal algorithm, topological sorting and shortest path.

CO5: to understand string matching, NP Completeness, Approximation.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF  
PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Introduction to Algorithms, The MIT Press Cambridge Thomas H. Cormen
2. Algorithm Design and Analysis, Oxford University, Harsh Bhasin, October 2015.
3. “Data Structures Through C in Depth” by Deepali Srivastava and S K Srivastava.

**Reference Books**

1. “Problem Solving in Data Structures and Algorithms Using C: The Ultimate Guide to Programming Interviews” by Hemant Jain.
2. “Data Structures and Algorithms Made Easy : Second Edition: Data Structure and Algorithmic Puzzles” by Narasimha Karumanchi
3. “Data Structures and Program Design In C” by Robert L Kruse and Bruce P Leung

**Course Name: Business Communication Skills**

**Course Code: MCA123B**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
2	0	0	2

**Course Objectives**

1. To apply an ability to develop communication theories and be highly skilled in the use of quantitative methods to evaluate them.
2. To apply ability to how people create, transmit, interpret, evaluate and respond to messages to inform, relate to, and influence one another interpersonally, in small groups, in organizations, in public settings and across cultures.
3. To build a firm knowledge of communication so as to facilitate its application for employability skills.
4. To enhance employability skills and facing the corporate world with full confidence.
5. To enhance the report writing and technical writing skills

**Syllabus**

**Unit I**

Basics of Communication

1. Introduction: What is Communication?
2. The Process of Communication : sender, receiver, channel, code, topic, message, context, feedback, 'noise'
3. Filters & Barriers to Communication
4. Different Types of Communication
5. The importance of communication
6. Verbal and non-verbal communication
7. Comparing general' communication and business communication

**Unit II**

Composition

1. Resume Writing
2. Job application

### **Unit III**

#### **Written Communication**

1. Report Writing
  - i. Definition and characteristics of report,
  - ii. Need of reports
  - iii. Types of reports: Technical Reports, Progress report, ,Event reporting,
  - iv. Newsletters
  - v. Summer project report
2. Technical proposal writing
  - i. Definition and characteristics of technical proposal writing,
  - ii. Types of proposal
  - iii. Making a proposal

### **Unit IV**

#### **Speaking Skills**

1. Seminar Presentation. :
  - i. Verbs often Required in Presentations
  - ii. Importance of Body Language in Presentations
  - iii. Preparing an Outline of a Presentation Pronunciation
  - iv. Structure of Presentation
  - v. Ending the Presentation
2. Group Discussion.
  - i. Definition,
  - ii. Advantages,
  - iii. Qualities Required,
  - iv. General Dos and Don'ts,
  - v. Body languages,
  - vi. Acting as a moderator

3. Interview:
  - i. Types of interview,
  - ii. Getting ready for an interview,
  - iii. Interview phases

## **Unit V**

1. Written Analysis of Cases/Case study
2. The research paper/the process of research

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

**Upon successful completion of this subject students should be able to:**

CO1: Understand communication processes and practices in professional and academic contexts.

CO2: Research and use information for communicative tasks.

CO3: Enhance employability skills and facing the corporate world with full confidence.

CO4: Enhance ability to develop communication theories and be highly skilled.

CO5: Enhance ability to write research papers and thesis.

### **MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books**

1. English Grammar and Composition, Gurudas Mukherjee, Ane Books Pvt. Ltd.(New Delhi)
2. Communication Skills for Engineers and Scientists, Sangeeta Sharma and BinodMishra, PHI Learning Pvt. Ltd.(New Delhi)

**Reference Books**

1. Current English Grammar and Usage with Composition, R.P. Sinha, Oxford University Press (New Delhi)
2. Effective Technical Communication, M Ashraf Rizvi, Tata McGraw Hill (New Delhi)
3. Business Communication, Meenakshi Raman & Prakash Singh, Oxford University Press (New Delhi)

**Course Name: Programming in Python Lab**

**Course Code: MCA147B**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

**Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Experiment**

**I. Exercise programs on basic control structures & loops.**

- a) Write a program for checking the given number is even or odd.
- b) Using a for loop, write a program that prints the decimal equivalents of 1/2, 1/3, 1/4 ,..... 1/10
- c) Write a program for displaying reversal of a number.
- d) Write a program for finding biggest number among 3 numbers.
- e) Write a program using a while loop that asks the user for a number, and prints a Countdown from that number to zero.

**II. Exercise programs on operators & I/O operations.**

- a) Write a program that takes 2 numbers as command line arguments and prints its sum.
- b) Implement python script to show the usage of various operators available in python language.
- c) Implement python script to read person's age from keyboard and display whether he is eligible for voting or not.
- d) Implement python script to check the given year is leap year or not.

**III. Exercise programs on Python Script.**

- a) Implement Python Script to generate first N natural numbers.
- b) Implement Python Script to check given number is palindrome or not.
- c) Implement Python script to print factorial of a number.
- d) Implement Python Script to print sum of N natural numbers.
- e) Implement Python Script to check given number is Armstrong or not.
- f) Implement Python Script to generate prime numbers series up to n

#### **IV. Exercise programs on Lists.**

- a) Finding the sum and average of given numbers using lists.
- b) To display elements of list in reverse order.
- c) Finding the minimum and maximum elements in the lists.

#### **V. Exercise programs on Strings.**

- a) Implement Python Script to perform various operations on string using string libraries.
- b) Implement Python Script to check given string is palindrome or not.
- c) Implement python script to accept line of text and find the number of characters, number of vowels and number of blank spaces in it.

#### **VI. Exercise programs on functions.**

- a) Define a function max\_of\_three() that takes three numbers as arguments and returns the largest of them.
- b) Write a program which makes use of function to display all such numbers which are divisible by 7 but are not a multiple of 5, between 1000 and 2000.

#### **VII. Exercise programs on recursion & parameter passing techniques.**

- a) Define a function which generates Fibonacci series up to n numbers.



- b) Define a function that checks whether the given number is Armstrong
- c) Implement a python script for Call-by-value and Call-by-reference
- d) Implement a python script for factorial of number by using recursion.

### **VIII. Exercise programs on Tuples.**

- a) Write a program which accepts a sequence of comma-separated numbers from console and generate a list and a tuple which contains every number. Suppose the following input is supplied to the program: 34, 67, 55, 33, 12, 98. Then, the output should be: ['34', '67', '55', '33', '12', '98'] ('34','67', '55', '33', '12', '98').
- b) With a given tuple (1, 2, 3, 4, 5, 6, 7, 8, 9, 10), write a program to print the first half values in one line and the last half values in one line.

### **IX. Exercise programs on files.**

- a) Write Python script to display file contents.
- b) Write Python script to copy file contents from one file to another.

### **X. Exercise programs on searching & sorting Techniques.**

- a) Implement a python script to check the element is in the list or not by using Linear search & Binary search.
- b) Implement a python script to arrange the elements in sorted order using Bubble, Selection, Insertion and Merge sorting techniques.

### **XI. Exercise programs on Python Libraries**

- a) Pandas, Numpy and Scipy
- b) Matplotlib and Scikit Learn

**Course Name: Programming in C++ Lab**

**Course Code: MCA125A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

**Course Objective**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Sample Programs**

Following are the list of sample programs. Students have to perform accordingly.

1. Write a program to calculate the distance between two points.
2. Write a program to perform addition, subtraction, multiplication and division on 2 integer numbers.
3. Write a program to swap 2 numbers using a temporary variable.
4. Write a program that displays the size of every data type.
5. Write a program to convert an integer into the corresponding floating point number.
6. Write a program to enter any character. If the entered character is in lower case, convert it into upper case. If it is a lower case character, then convert it into upper case.
7. Write a program to find whether a given year is leap year or not.
8. Write a program to print 20 horizontal asterisks (\*)
9. Write a program to classify a given number as prime or composite.
10. Write a program to sum the series:  $1/1+2^2/2+3^3/3+....$
11. Write a program to print the following pattern:

```
A
AB
ABC
ABCD
ABCDE
```

12. Write a program to print the following pattern:

```
1
121
12321
1234321
```

13. Write a program using for loop to calculate the average of first n natural numbers.

14. Write a program to enter a decimal number. Calculate and display the binary equivalent of this number.

15. Write a program to display the largest of 10 numbers using ternary operator.

16. Write a program to generate calendar of a month given the start day and the number of days in that month.

17. Write a program to calculate the area of a circle using functions.

18. Write a program to calculate the volume of a cuboid using default arguments.

19. Write a program to add two values of different data types using static polymorphism.

20. Write a program to calculate GCD using recursive function.

21. Write a program to calculate  $\exp(x,y)$  using recursive functions.

22. Write a program to read and display n random numbers using an array.

23. Write a program to interchange the largest and the smallest number in the array.

24. Write a program to find the second largest number using an array of n numbers.

25. Write a program to merge 2 unsorted arrays.

26. Write a program to implement linear search in any given array.

27. Write a program to transpose a 3\*3 matrix.

28. Write a program to read and display a 2\*2\*2 array.

29. Write a program to concatenate 2 strings.

30. Write a program to compare 2 strings.

31. Write a program to insert a string in the main text.

32. Write a program to sort names of students.

33. Write a program to enter a text that contains multiple lines. Display the n lines of text starting from m<sup>th</sup> line.

34. Write a program to add 2 integers using pointer.

35. Write a program to add 2 vectors (Arrays).

36. Write a program to copy a given string into a new string. Memory for the new string must be allocated dynamically.
37. Write a program to read, display, add and subtract 2 complex numbers.
38. Write a program to read and display information of a student using structure within a structure.
39. Write a program to read and display information of all the students in the class.
40. Write a program using pointer to structure to initialize the members in the structure.
41. Write a program to illustrate the use of arrays within structure.
42. Write a program to display the name of the colors using enumerated types.
43. Write a program to enter a rational number, simplify and display it.
44. Write a program to demonstrate the concept function overloading in the member functions of a class.
45. Write a program that adds 2 complex numbers. The object must be passed through reference and the result must be passed by value.
46. Write a program to display the list of students according to their marks.
47. Write a program to find mean of 2 numbers belonging to two different classes using friend function.
48. Write a program to add 2 arrays using friend function and operator overloading.
49. Write a program that uses an overloaded constructor to dynamically allocate memory to an array and thus find the largest of its elements.
50. Write a program that uses dynamic constructor to allocate memory to an array. Count the number of even and odd elements.
51. Write a program to overload the increment and decrement operators on the array class.
52. Write a program to find the next date using unary increment operator.
53. Write a program to demonstrate the use of type conversions with distance class.
54. Write a program that uses multi level inheritance to implement classes- note, server and device.
55. Write a program with class bill. The users have the option to pay the bill either by cheque or by cash. Use the inheritance to model this situation.

**Course Name: Advance Data Structure and Algorithm Lab**

**Course Code: MCA126A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

**Course Objective**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Sample Programs**

1. Write a program for performing Push & Pop operations in Stack.
2. Write a program for performing Insertion & Deletion operations in Circular Queue.
3. Write a program for Merge Sort.
4. Write a program for Quick Sort.
5. Write a program to demonstrate any Hash Function.
6. Write a program for creating a Binary Search tree and taking its Preorder, Postorder and Inorder.
7. Write a program for identifying that given binary tree is complete binary tree or not.
8. Write a program to create a maximum priority queue using max heap tree.
9. Write a program to create AVL tree.
10. Write a program for identifying that given graph is complete graph or not.
11. Write a program for Depth First Search algorithm in Graph.
12. Write a program for Breadth First Search algorithm in Graph.
13. Write a program for Topological Sort.
14. Write a program of maintaining a Spanning Tree using Kruskal or Prim's Algorithm.
15. Write a program for finding Shortest Paths from given graph using Dijkstra's Algorithm.

**Course Name: Business Communication Skills Lab**

**Course Code: MCA127B**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

**Course Objective**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Sample Programs**

1. Introduction of Phonetics
2. Pronunciation Exercise
3. Describing the Pictures or images and cartoon using MSWord
4. Developing outline, key expression and situation
5. Jumbled words/sentences
6. Group Discussion
7. Job inter view
8. Seminar Presentation
9. Resume Writing Practicing

## **Semester II**

**Course Name: Advance Java**

**Course Code: MCA130A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	0	0	3

### **Course Objectives**

1. To exhaustive coverage of advanced topics on Java from tools to enterprise Java
2. To provide ample application-based examples, with step-by-step explanations
3. To provide thorough understanding of each topic through extensive examples along with the program codes and screenshots
4. To provides relevant software installation and configuration information wherever necessary
5. To comprises keywords, objective-type questions (with answers) and subjective-type questions for students at the end of all the chapters

### **Syllabus**

#### **Unit I**

**Introduction** : Java Evolution and history, Data types, Control statements , conditional statements, Classes and Objects, Arrays and strings, Functions & Interfaces, Inheritance, keywords: Static, Final, Super, Packages,

#### **Unit II**

**Applet, Exception Handling, Multi-threading & Garbage Collection:** Applet, Applet life Cycle, Exception Handling: Introduction, types, catching exceptions, tracing stack, custom exception classes

Multi-threading : Introduction, Main Thread, Creating Thread, Interrupting Thread, Suspending and Resuming, Thread Priority, Garbage Collection.

#### **Unit III**

**Collection frame work and Generic Programming:** Collection frame work: Introduction, Benefits, Collection Interfaces, and Collection Implementation.

Generic Programming: Introduction, Collection Framework and Generics, Type Naming, Generic Methods and Constructors, Type Inference, Bounded Type Parameters, Wildcards, Type Erasure, Restrictions on Generics.

#### **Unit IV**

**AWT, Swings & Input/Output:** AWT: AWT Class Hierarchy, Creating Container, Adding Components, Layout, AWT components, Event Handling, Dialog Boxes, Scrollbar, Menu.

Swings: Containment Hierarchy, Swing Components, Methods of Important Event Listener Interfaces Streams, Formatting, Data Streams, Object Stream, Reading/writing Arrays via Streams, Pipes, File I/O, Path, File

#### **Unit V**

**JDBC, Servlet & JSP:** JDBC: JDBC Drivers, JDBC Architecture, JDBC Classes and Interfaces, Loading a Driver, Making a Connection, Execute SQL Statement, SQL Statements, Retrieving Result, Getting Database Information, Metadata.

Servlet: Server-side Java, Servlet Architecture, Servlet Life Cycle, GenericServlet.

JSP: JSP and HTTP, JSP Engines, JSP and Servlet, JSP Syntax, JSP Components

#### **Course Outcomes (Cos):**

**Upon successful completion of this subject students should be able to:**

CO1: Revise object oriented features of java language and develop java applet programming using various techniques.

CO2: Handling exceptions and develop multi- threaded applications.

CO3: Develop applications using collection framework and concepts of generic programming.

CO4: Develop applications using Abstract Window Toolkit

CO5: Develop server side programs using Servlets and develop Java Server Pages applications using JSP Tags.



**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Advance java programming, Oxford, Uttam Kumar Roy, April 2015.
2. Sachin Malhotra, Saurabh Chaudhary, "Programming in Java", Second Edition, Oxford University Press , 2014.

**Reference Books**

1. Programming with Java A Primer, E.Balaguruswamy Tata McGraw Hill Companies
2. Java Programming Language Ken Arnold Pearson
3. The complete reference JAVA2, Herbert schildt. TMH

**Course Name: PHP & MYSQL**

**Course Code: MCA132B**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	0	0	3

**Course Objectives**

1. To demonstrate the basics of the PHP and MYSQL, PHP configuration in IIS & Apache Web Server and features of PHP.
2. To understand how PHP, HTML and MYSQL work together to produce dynamic pages.
3. To apply specific PHP variables, data types, syntax and conditional statements.
4. To be able to apply control statements and database connectivity.
5. To justify the database connectivity of PHP with MYSQL.

**Syllabus**

**Unit I**

Introduction to Dynamic Web Content, What Is a WAMP, MAMP, or LAMP, Installing WAMP on Windows, Introduction to PHP, Incorporating PHP within HTML, The Structure of PHP, Using Comments, Basic Syntax, Variables, Operators ,Variable Assignment ,Multiple-Line Commands, Variable Typing ,Constants, Predefined Constants ,The Difference Between the echo and print Commands.

**Unit II**

Functions ,Variable Scope. Expressions, Operators, Conditionals, Looping, PHP Functions, PHP Objects, PHP Arrays, Using Array Functions, Date and Time Functions. File Handling- Checking Whether a File Exists, Creating a File, Reading from Files, Copying Files, Moving a File ,Deleting a File ,Updating Files ,Locking Files for Multiple Accesses ,Reading an Entire File ,Uploading Files, Building Forms, Retrieving Submitted Data.

### **Unit III**

Using Cookies in PHP, Setting a Cookie ,Accessing a Cookie, Destroying a Cookie, HTTP Authentication, Storing Usernames and Passwords, Using Sessions, Starting a Session, Ending a Session, Setting a Timeout, Session Security.

### **Unit IV**

Introduction to MySQL,MySQL Basics, Summary of Database Terms, Accessing MySQL via the Command Line, Starting the Command-Line Interface, MySQL Commands,Querying a MySQL Database, Joining Tables Together ,Using Logical Operators,MySQL Functions , Accessing MySQL via phpMyAdmin ,Database Design, Primary Keys: The Keys to Relational, Databases ,Normalization, Relationships ,Transactions ,Creating a Backup File , Querying a MySQL Database with PHP, Creating a Login File, Connecting to a MySQL Database, The \$\_POST Array, Creating a Table, Describing a Table ,Dropping a Table ,Adding Data, Retrieving Data, Updating Data ,Deleting Data ,Using AUTO\_INCREMENT.

### **Unit V**

JavaScript and HTML,Using Scripts Within a Document Head, Including JavaScript Files ,Using Comments, Semicolons, Variables, String Variables, Numeric Variables, Arrays, Operators ,The Document Object Model, Using the DOM, About document.write, Using alert, Functions,Global Variables, Expressions and Control Flow in JavaScript, Expressions ,Conditionals statements ,Looping ,JavaScript and PHP Validation ,Validating User Input with JavaScript , Redisplaying a Form After PHP Validation.

### **Course Outcomes:**

**Upon successful completion of this subject students should be able to:**

CO1: Demonstrate the basics of the PHP and MYSQL, PHP configuration in IIS & Apache Web Server and features of PHP,basics of PHP.

CO2: Examine how web pages are developed using PHP array, form and input elements.

CO3: Introduction to MySQL,MySQL Basics, MySQL Commands.

CO4: JavaScript and HTML,Using Scripts and PHP Validation ,Validating User Input with JavaScript

CO5: Using Ajax, What Is Ajax, Introduction to CSS,Importing a Style Sheet

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF  
PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

<b>Course Outcomes</b>	<b>Program Outcomes</b>											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1				M				M				
CO2		M				M					M	
CO3							M					
CO4			H						M			L
CO5	M					L						

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Learning PHP, MySQL & JavaScript With jQuery, CSS & HTML5- by Robin Nixon(Published by-O'Reilly)
2. PHP5 and MySQL®Bible- Wiley Publishing

**Reference Books**

1. Core PHP Programming, Leon Atkinson Pearson publishers, 2nd Edition.
2. PHP Cookbook by David Sklar and Adam Trachtenberg, O'Reilly Media, Inc., ISBN: 978-1-449-36375-8.
3. The Complete Reference PHP, SteverHolzner McGraw Hill, Edition: 1 ISBN: 9780070223622.

4. Web Technologies: HTML, JAVASCRIPT, PHP, JAVA, JSP, ASP.NET, XML and Ajax, Black Book: HTML, Javascript, PHP, Java, Jsp, XML and Ajax, Black Book” by Kogent Learning Solutions Inc.

**Course Name: Advance Database Management System**

**Course Code: MCA118A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	0	0	3

### **Course Objectives**

1. To enhance the fundamentals knowledge of data models and to conceptualize and depict a database system using ER diagram.
2. To know fundamentals of Operations of Relational Algebra and calculus.
3. To know the fundamental concepts of normalization.
4. To justify the concept of transaction processing management, concurrency control techniques and recovery procedure.
5. To have an introductory knowledge about the Storage and Query processing Techniques.

### **Syllabus**

#### **Unit I**

**Data modeling:** Entity Relationship Model, Relational, Network, Hierarchical and object oriented models, Data Modeling using the Entity Relationship Model. Relational Constraints, Domain Constraints, Key Constraints Referential Integrity Constraints, Relational Algebra and Relational Calculus.

#### **Unit II**

**Database Design:** Integrity Constraints – Domain Constraints- Referential integrity – Functional Dependency- Normalization using Functional Dependencies, Normal forms based on primary keys- general definitions of Second and Third Normal Forms. Boyce Codd Normal Form– Multivalued Dependencies and Forth Normal Form – Join Dependencies and Fifth Normal Form.

### **Unit III**

**Object Relational Databases:** Complex Data Types and Object Orientation, Structured Data Types and Inheritance in SQL, Table Inheritance, Array and Multiset Types in SQL, Object Identity and Reference Types in SQL, Implementing O-R Features, Persistent Programming Languages, Comparison of Object-Oriented and Object-Relational Database

### **Unit IV**

**Physical Database Design:** Overview of Physical Storage Media, Magnetic Disks, RAID, Tertiary Storage , Storage Access, File Organization, Organization of Records in Files, Data-Dictionary Storage, Storage Structures for Object-Oriented Databases, Basic Concepts, Ordered Indices , B<sup>+</sup>-Tree Index Files, B-Tree Index Files, Static Hashing, Dynamic Hashing , Comparison of Ordered Indexing and Hashing , Index Definition in SQL.

### **Unit V**

**Transaction Management:** Transaction Concept, ACID Properties, Transaction State, Implementation of ACID properties, Schedules and Serializability: Conflict Serializability, View Serializability. Concurrency Control: Need of concurrency control, Concurrency control techniques, Lock based protocols, binary lock, share and exclusive lock, two phase locking protocol. Introduction to recovery.

### **Course Outcomes (Cos)**

**After successfully completing this subject, students will be able to:**

CO1: Understand practical implications of transaction properties and concurrency control techniques.

CO2: Understand the fundamentals of Object Relational database and complex data types.

CO3: Gain about the fundamentals of physical storage media and indexing.

CO4: Enhance the fundamentals knowledge of data models and to conceptualize and depict a database system using ER diagram.

CO5: Contrast the concept of functional dependency, Norm forms, constraints and integrity

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books**

1. RamezElmasri, Shamkant B. Navathe, “Fundamentals of Database Systems”, Fifth Edition, Pearson, 2008
2. A.Silberschatz, H. Korth and S. Sudarshan, Database System Concepts, 5th Edition, McGraw Hill.

**Reference Books**

1. C.J.Date, A.Kannan, S.Swamynathan, “An Introduction to Database Systems”, Eighth Edition, Pearson Education, 2006.
2. .
3. Raghu Ramakrishnan, “Database Management Systems”, Fourth Edition, Tata McGraw Hill, 2010.
4. G.K.Gupta,”Database Management Systems”, Tata McGraw Hill, 2011.

**Course Name: Advance Java Lab**

**Course Code: MCA133A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

### **Course Objectives**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

### **List of Sample Programs**

1. Write a Java program that prints all real solutions to the quadratic equation  $ax^2+bx+c = 0$ . Read in a, b, c and use the quadratic formula. If the discriminate  $b^2-4ac$  is negative, display a message stating that there are no real solutions?
2. The Fibonacci sequence is defined by the following rule. The first 2 values in the sequence are 1, 1. Every subsequent value is the sum of the 2 values preceding it. Write a Java program that uses both recursive and non-recursive functions to print the nth value of the Fibonacci sequence?
3. Write a Java program that prompts the user for an integer and then prints out all the prime numbers up to that Integer?
4. Write a Java program that checks whether a given string is a palindrome or not. Ex: MADAM is a palindrome?
5. Write a Java program that reads a line of integers and then displays each integer and the sum of all integers.
6. To write a java program to show how a class implements two interfaces.
7. To write a java program to show that the variables in an interface are implicitly static and final and methods are automatically public.
8. Write a program to implement multilevel inheritance and method overriding.



9. Write a program to illustrate class member access for packages and also implement interfaces.
10. To write a java applet program to change the color of a rectangle using scroll bars to change the value of red, green and blue
11. To write an applet program for creating a simple calculator to perform Addition, subtraction, Multiplication and Division using Button, Label and TextField component.
12. Write a program in Java to describe concepts of handling exceptions.
13. Write a program in Java to describe Exception Handling with all keywords
14. Write a program to describe the concept of Collection framework - an ArrayList.
15. Write a program to describe Java Generics using Map.
16. Write a program in Java in AWT to design login form.
17. Write a Java program to draw a rectangle using swing.
18. Write a Java swing program to print a wave form with output.
19. Write a Enable and Disable button in Java swing.
20. Write a java program that connects to a database using JDBC and does add, deletes, modify and retrieve operations.
21. Write a program to update student information using jdbc connection.
22. Write a program to demonstrate dynamic HTML using java servlet.
23. Write a program to create an applet of a moving banner.
24. Develop static pages (using only HTML) of an online Book store.
25. Validate the registration, user login, user profile and payment by credit card pages using Java servlet.
26. Write a program to demonstrate basic servlet.
27. Write a program to demonstrate basic JSP example.
28. Write a program to perform database operations in JSP.
29. Write a program to create session management in JSP.
30. Create a Cookie and add these four user id's and passwords to this Cookie using servlet.
31. Write a Servlet code to implement the session handling in java.
32. Write a JSP code which does the following job: Insert the details of the 3 or 4 users who register with the web site by using registration form. Authenticate the user when he submits the login form using the user name and password from the database.

33. Develop a simple website using JSP and servlet , session handling , login

**Course Name: PHP & MYSQL Lab**

**Course Code: MCA134B**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

### **Course Objectives**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

### **List of Sample Programs**

1. WAP to reverse the string without using any predefined function
2. WAP to check the season of the year according to given month using control statement(Switch statement)
  - a. spring month 3,4,5
  - b. summer month 6,7,8
  - c. autumn month 9,10,11
  - d. winter month 12,1,2
3. WAP to create an array for employee names and associate another array of employee's salary and print these values using for each loop.
4. WAP that will accept an array of integers as input and output an array where for each item in the source array, the new array will perform the following operations:-
  - a. for even numbers divide by 2
  - b. for odd number multiply by 3
5. WAP to make sure a value has been supplied for a form element. For example, you want to make sure a text box hasn't been left blank.
6. WAP to know whether an email address a user has provided is valid.
7. WAP in php to show the values in the input fields after the user hits the submit button

8. WAP create a form with one text field and submit button to find string in lower case, uppercase, sub string, position of a string and trimmed string
9. WAP to find whether a given string is subset of another given string or not
10. WAP to find square and square root of a randomly generated number
11. WAP to merge 2 associative array sort that array in :-
  - a. ascending order
  - b. descending order
  - c. in key sorted form
12. WAP to generate captcha code
13. WAP to create session, provide it name and id and display their values and finally delete the session.
14. WAP to in which you want to set a cookie so that your website can recognize subsequent requests from the same web browser.
15. WAP to in which you want to read the value of a cookie that you've previously set.
16. WAP in which You want direct access to the body of a request, not just the parsed data that PHP puts in \$\_POST for you. For example, you want to handle an XML document that's been posted as part of a web services request.
17. WAP to shows how to use validate().
18. WAP to Call ob\_start() at the top of your page and ob\_end\_flush() at the bottom. You can then intermix commands that generate output and commands that send headers.
19. WAP to send compressed content to browsers that support automatic decompression.
20. WAP to use the same HTML page to emit a form and then process the data entered into it. In other words, you're trying to avoid a proliferation of pages that each handle different steps in a transaction.
21. WAP in which you have data that can be easily represented as key/value pairs, want to store it safely, and have very fast lookups based on those keys.
22. WAP in which you want access to a SQL database to store or retrieve information. Without a database, dynamic websites aren't very dynamic.
23. WAP in which you want to retrieve some data from your database.
24. WAP in which you want a concise way to execute a query and retrieve the data it returns.

25. WAP in which you want to know how many rows a SELECT query returned or you want to know how many rows an INSERT, UPDATE, or DELETE query changed.
26. WAP in which you need to make text or binary data safe for queries.
27. Write a JavaScript program to display the current day and time in the following format.
28. Write a JavaScript program to find the area of a triangle where lengths of the three of its sides are 5, 6, 7.
29. Write a JavaScript program to calculate multiplication and division of two numbers (input from user).
30. Write a JavaScript program to convert temperatures to and from Celsius, Fahrenheit.
31. Write a JavaScript function that reverse a number
32. Write a JavaScript function that returns a passed string with letters in alphabetical order.
33. Write a JavaScript function that accepts a string as a parameter and counts the number of vowels within the string.
34. Write a JavaScript conditional statement to find the largest of five numbers. Display an alert box to show the result.
35. Write a JavaScript for loop that will iterate from 0 to 15. For each iteration, it will check if the current number is odd or even, and display a message to the screen.

**Course Name: Minor Project**

**Course Code: MCA136A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

**Course Objectives**

The purpose of minor project is to enhance the practical knowledge based on prescribed languages which are introduced in this program. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

### **Semester V**

**Course Name: Accounting Principles and Practices**

**Course Code: MCA129A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	0	0	3

#### **Course Objectives:**

1. To be acquainted with about the important concepts and characteristics of accounting.
2. To study the application of accounting in the general business environment.
3. To prepare the estimate for various business activities such as purchase, sale, production and cash budgets.
4. To provide a comprehensive treatment of accounting principles, technique and practices.
5. To get the students acquainted with fundamental concepts and processes of accounting so that they are able to appreciate the nature of item presented in the annual accounts of an organization.

#### **Syllabus**

##### **Unit I**

**Accounting Concept:** Basic of Accounting, Meaning and nature of accounting, Scope of financial accounting, Interrelationship of Accounting with other disciplines, Branches of Accounting, Accounting concepts and convention, Indian Accounting Standards

##### **Unit II**

**Journal, Ledger and Trial Balance:** Journal: Meaning of Journal, Advantages, and Subdivision. Ledger: Meaning, subdivision, Mechanics of Posting, balancing of Ledger accounts. Trial Balance: Objectives, Defects of trial balance, Errors disclosed by trial balance, preparation and locating errors.

### **Unit III**

**Cash Book and Subsidiary books of Accounting:** Kinds of cashbook, Purchase daybook, Sales daybook, Bills receivable book, Bills payable book.

### **Unit IV**

**Finance Accounts:** Trading account, Profit & Loss account, Adjustments, Balance Sheet, Forms of balance Sheet, Assets and their classification, liabilities and their classification, uses and limitations.

### **Unit V**

**Depreciation:** Causes of Depreciation, Meaning of Depreciation Accounting, Method of Recording depreciation, Methods of Providing Depreciation.

### **Course Outcome:**

**Upon successful completion of this subject students should be able to:**

CO1: Get the Knowledge about the important concepts and characteristics of accounting.

CO2: Able to understand the budget preparation and control of a company.

CO3: Be prepared of fiscal policies of the organization.

CO4: Decide about the state of affairs of a particular firm/company.

CO5: Understand the concept of causes of depreciation.

### **MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

Course	Program Outcome
--------	-----------------

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

H = Highly Related; M = Medium; L = Low

### Text Books

1. Tulsian, P.C., (2009) Financial Accountancy, 2nd edition, Pearson Education.
2. Maheshwari, S.N. and Maheshwari, S. K., (2009) An Introduction to Accountancy, Eighth Edition, Vikas Publishing House.

### Reference Books

1. Essentials of Financial Accountng (based on IFRS), Bhattacharya (PHI, 3rd Ed) 3)
2. Ramachandran Kakani- Financial Accounting for Management( TMH ,3rd Edition). 4)
3. Dhamija - Financial Accounting for managers: (Prentice Hall, 2nd Edition).



**Course Code: MCA183A**  
**Course Name: Agile Software development**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>C</b>
3	0	0	3

**Course Objectives:**

1. To provide students with a theoretical as well as practical understanding of agile software development practices and how small teams can apply them to create high-quality software.
2. To provide a good understanding of software design and a set of software technologies and APIs.
3. To do a detailed examination and demonstration of agile development and testing techniques.
4. To understand the benefits and pitfalls of working in an agile team.
5. To understand agile development and testing.

**UNIT I**

**Agile Methodology:-** Theories for Agile Management – Agile Software Development – Traditional Model vs. Agile Model – Classification of Agile Methods – Agile Manifesto and Principles – Agile Project Management – Agile Team Interactions – Ethics in Agile Teams – Agility in Design, Testing – Agile Documentations – Agile Drivers, Capabilities and Values

## **UNIT II**

**Agile Processes:** - Lean Production – SCRUM, Crystal, Feature Driven Development- Adaptive Software Development – Extreme Programming: Method Overview – Lifecycle – Work Products, Roles and Practices.

## **UNIT III**

**Agility And Knowledge Management:-** Agile Information Systems – Agile Decision Making – Earl\_S Schools of KM – Institutional Knowledge Evolution Cycle – Development, Acquisition, Refinement, Distribution, Deployment , Leveraging – KM in Software Engineering – Managing Software Knowledge – Challenges of Migrating to Agile Methodologies – Agile Knowledge Sharing – Role of Story-Cards – Story-Card Maturity Model (SMM).

## **UNIT IV**

**Agility and Requirements Engineering:-** Impact of Agile Processes in RE–Current Agile Practices – Variance – Overview of RE Using Agile – Managing Unstable Requirements – Requirements Elicitation – Agile Requirements Abstraction Model – Requirements Management in Agile Environment, Agile Requirements Prioritization – Agile Requirements Modeling and Generation – Concurrency in Agile Requirements Generation.

## **UNIT V**

**Agility and Quality Assurance:** - Agile Product Development – Agile Metrics – Feature Driven Development (FDD) – Financial and Production Metrics in FDD – Agile Approach to Quality Assurance – Test Driven Development – Agile Approach in Global Software Development.

### **Course Outcomes:**

**Upon completion of the course, the students will be able to:**

CO1 Realize the importance of interacting with business stakeholders in determining the requirements for a software system

CO2 Perform iterative software development processes: how to plan them, how to execute them.

CO3 Point out the impact of social aspects on software development success.

CO4 Develop techniques and tools for improving team collaboration and software quality.

CO5 Perform Software process improvement as an ongoing task for development teams and show how agile approaches can be scaled up to the enterprise level.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books:**

1. David J. Anderson and Eli Schragenheim, —Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results, Prentice Hall, 2003.
2. Hazza and Dubinsky, —Agile Software Engineering, Series: Undergraduate Topics in Computer Science, Springer, 2009.

**REFERENCES:**

1. Craig Larman, —Agile and Iterative Development: A Managers Guide, Addison-Wesley, 2004.

2. Kevin C. Desouza, —Agile Information Systems: Conceptualization, Construction, and Management, Butterworth-Heinemann, 2007.

**Course Name: Information Security & Cyber Law**

**Course Code: MCA138A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	0	0	3

### **Course Objectives**

1. To learn today's increasing network security threats and explain the need to implement a comprehensive security policy to mitigate the threats.
2. To provide extended security using authentication, Substitution Techniques, Transposition Techniques, Encryption and Decryption
3. To introduce security services for email and email protocols, Digital Certificates, Private Key Management, E-mail Security and Wireless Application Protocol (WAP) Security
4. To be aware about Prohibited actions on Cyber, Cyber Squatting Banking/Credit card related crime E-commerce.
5. To gain the knowledge about Cyber Crime and Prohibited actions on Cyber.

### **Syllabus**

#### **Unit I**

**Introduction to the Concepts of Security:** The need for security, Security Approaches, Principles of Security, Types of Attacks. Cryptographic Techniques: Plain Text and Cipher Text, Substitution Techniques, Transposition Techniques, Encryption and Decryption, Symmetric and Asymmetric Key Cryptography, Key Range and Key Size, Possible Types of Attacks.

## **Unit II**

**Computer-based Symmetric Key Cryptographic Algorithms:** Algorithm Types and Modes, An overview of Symmetric Key Cryptography, DES, International Data Encryption, Algorithm (IDEA), RC5, Blowfish, AES, Differential and Linear Cryptanalysis.

## **Unit III**

**Computer-based Asymmetric Key Cryptography:** Brief History of Asymmetric Key Cryptography, An overview of Asymmetric Key Cryptography, The RSA Algorithm, Symmetric and Asymmetric Key Cryptography Together, Digital Signatures, Knapsack Algorithm, Some other Algorithms.

## **Unit IV**

**Public Key Infrastructure:** Digital Certificates, Private Key Management, The PKIX Model, Public Key Cryptography Standards, XML, PKI and Security. Internet Security Protocols: Basic Concepts, Secure Socket Layer, SHTTP, Time Stamping Protocol, Secure Electronic Transaction, SSL versus SET, 3-D Secure Protocol, Electronic Money, E-mail Security, Wireless Application Protocol (WAP) Security, Security in GSM.

## **Unit V**

**Prohibited Actions on Cyber:** Pornography, IPR violations: software piracy, copyright infringement, trademarks violations, theft of computer source code, patent violations, Cyber Squatting Banking/Credit card Related crime E-commerce/ Investment Frauds, Defamation (Cyber smearing),Cyber Stacking

## **Course Outcomes (Cos):**

**Upon successful completion of this subject students should be able to:**

CO1: Demonstrate the threats in networks and security concepts, Plain Text and Cipher Text, Substitution Techniques, Transposition Techniques, Encryption and Decryption.

CO2: Apply authentication applications in different networks.

CO3: Understand security services for email, the RSA Algorithm, Symmetric and Asymmetric Key Cryptography Together and Digital Signatures.

CO4: Know of Digital Certificates, Private Key Management, E-mail Security and Wireless Application Protocol (WAP) Security

CO5: Awareness of Prohibited actions on Cyber, Cyber Squatting Banking/Credit card related crime E-commerce.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books**

1. William Stallings, —Cryptography and Network Security, Prentice Hall, New Delhi, 2006.
2. Atul Kahate, Cryptography and Network Security, Mc Graw Hill Education, 3<sup>rd</sup> Edition.

**Reference Books**

1. Neal Krawetz, —Introduction to Network Security, Thomson Learning, Boston, 2007.
2. Bruce Schneier, —Applied Cryptography, John Wiley & Sons, New York, 2004.
3. Frontiers of of Electronic Commerce Kalakota and Whinstn Addition Wesley

**Course Name: Data Science & Analytics**

**Course Code: MCA184A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
2	1	0	3

### **Course Objectives**

1. Able to explain Statistical Analysis, Python Environment Setup and working of Numpy.
2. Able to design Data Visualization in Python using matplotlib.
3. Able to perform linear and multiple linear regression analyses.
4. Ability to select and implement machine learning techniques in real life applications.
5. Learn about computing environment that is suitable for the applications under consideration.
- 6.

### **Syllabus**

#### **Unit - I**

Data Science Overview, Data Analytics Overview, Statistical Analysis and Business Applications, Python Environment Setup and Essentials, Mathematical Computing with Python – NumPy: - Introduction to Numpy, Creating and Printing an ndarray, Class and Attributes of ndarray, Basic Operations, Slicing, Mathematical Functions of Numpy.

#### **Unit- II**

Data Manipulation with Pandas: - Introduction to Pandas, Understanding DataFrame, View and Select Data, Missing Values, Data Operations, File Read and Write Support, Pandas Sql Operation, Analyse different Dataset using Pandas.

### **Unit- III**

Data Visualization in Python using matplotlib: - Introduction to Data Visualization, Line Properties, Plot and Subplots, Types of Plots, Draw a pair plot using seaborn library.

### **Unit- IV**

Scientific computing with Python (Scipy):- Introduction to SciPy, SciPy Sub Package - Integration and Optimization, SciPy sub package - Statistics, Weave and IO, Solving Linear Algebra problem using SciPy.

### **Unit- V**

Machine Learning with Scikit-Learn: Machine Learning Approach, Supervised Learning Model Considerations, Supervised Learning Models - Linear Regression, Logistic Regression, K Nearest Neighbours, Decision Tree: Regression and Classification Trees, Support Vector Machines, Unsupervised Learning Models.

### **Course Outcomes (Cos):**

CO1: Students will able to explain Statistical Analysis,Python Environment Setup and working of Numpy.

CO2: Students will able to Manipulate and analysis data with Pandas

CO3: Students will able to design Data Visualization in Python using matplotlib.

CO4: Students will be able to learn about scientific computing with Python

CO5: Students will be able to learn Machine Learning Approaches using Scikit Learn.

### **MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**



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

H = Highly Related; M = Medium; L = Low

### Text Books

1. Hastie, Trevor, et al., The elements of statistical learning. Vol. 2. No. 1. New York: springer, 2009.
2. Montgomery, Douglas C., and George C. Runger., Applied statistics and probability for engineers. John Wiley & Sons, 2010

**Course Name: Minor Project**

**Course Code: MCA143A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	4	4

### **Course Objectives**

The purpose of minor project is to enhance the practical knowledge based on prescribed languages which are introduced in this program. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

## **List of Courses for Elective – II & IV**

**Course Name: Parallel Computing**

**Course Code: MCA150A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

### **Course Objectives:**

1. To identify the scope for parallelism in a program and Routing Mechanisms for Interconnection Networks.
2. To understand the various parallel programming models and the challenges involved in parallel programming.
3. To learn the basics of heterogeneous systems using CUDA and OpenCL.
4. To introduce the foundations of parallel computing, including parallel architectures, parallel programming methods and techniques.
5. To introduce parallel algorithm designs, and parallel performance analysis.

### **Syllabus**

#### **Unit I**

**Introduction to Parallel Computing:** Motivating Parallelism, Scope of Parallel Computing, Organization and Contents of the Text, Parallel Programming Platforms, Implicit Parallelism: Trends in Microprocessor Architectures, Limitations of Memory System Performance, Dichotomy of Parallel Computing Platforms, Physical Organization of Parallel Platforms, Communication Costs in Parallel Machines, Routing Mechanisms for Interconnection Networks.

#### **Unit II**

**Principles of Parallel Algorithm Design:** Introduction, Decomposition Techniques, Characteristics of Tasks and Interactions, Mapping Techniques for Load Balancing, Methods for Containing Interaction Overheads, Parallel Algorithm Models

### **Unit III**

**Basic Communication Operations:** One-to-All Broadcast and All-to-One Reduction, All-to-All Broadcast and Reduction, All-Reduce and Prefix-Sum Operations, Scatter and Gather, All-to-All Personalized Communication, Circular Shift, Improving the Speed of Some Communication Operations

### **Unit IV**

**Analytical Modeling of Parallel Programs:** Sources of Overhead in Parallel Programs, Performance Metrics for Parallel Systems, The Effect of Granularity on Performance, Scalability of Parallel Systems, Minimum Execution Time and Minimum Cost-Optimal Execution Time, Asymptotic Analysis of Parallel Programs.

Programming Using the Message-Passing Paradigm: The Building Blocks: Send and Receive Operations, MPI: the Message Passing Interface, Topologies and Embedding

### **Unit V**

**Programming Shared Address Space Platform:** Thread Basics, The POSIX Thread API, Thread Basics: Creation and Termination, Synchronization Primitives in Pthreads, Controlling Thread and Synchronization Attributes, Thread Cancellation, Composite Synchronization Constructs

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

**Upon successful completion of this subject students should be able to:**

CO1: Able to understand the concept of parallelism and Routing Mechanisms for Interconnection Networks.

CO2: Able to understand the parallel algorithm design techniques.

CO3: To design the Analytical Models of Parallel Programs.

CO4: To understand the concept of thread, POSIX and Pthreads, Thread Cancellation and Composite Synchronization Constructs

CO5: To understand the basic communication operations, message passing interfaces and topologies.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Introduction to parallel Computing, Oxford University, Petersen and Arbenz, July 2008.
2. “Advanced Computer Architecture for Parallel Processing (Mcgraw Hill Series in Electrical and Computer Engineering)” by Kai Hwang

**Reference Books**

1. “Computer Architecture and Parallel Processing” by Kai Hwang
2. “Computer Architecture and Parallel Processing” by Bharat Bhushan Agarwal and Sumit Prakash Tayal.
3. “Parallel Computers – Architecture and Programming” by Rajaraman V

**Course Name: Advance Computer Architecture**

**Course Code: MCA151A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objective:**

1. To describe the operation of modern and high performance computers.
2. To undertake performance comparisons of modern and high performance computers.
3. To improve the performance of applications on modern and high performance computers.
4. To development of software to solve computationally intensive problems.
5. To enhance teamwork and leadership skills through the project.

**Syllabus**

**Unit I**

**Introduction to Advanced Computer Architecture and Parallel Processing:** Four Decades of Computing, Flynn's Taxonomy of Computer Architecture, SIMD Architecture, MIMD Architecture, Interconnection Networks

**Unit II**

**Performance Analysis of Multiprocessor Architecture:** Computational Models, an Argument for Parallel Architectures, Interconnection Networks Performance Issues, Scalability of Parallel Architectures

**Unit III**

**Instruction Set and Addressing Modes:** Pipelining, Categories of Instructions: General Instruction Format, General Addressing Modes, Program Control, An Example: the M68000 Addressing Modes, Instruction Set of MIPS Processor, Link to Subroutines, Three Instruction Formats, Three Ways of Branch/Jump Addressing, Three Addressing Modes for Operands, CISC vs. RISC

#### **Unit IV**

**Memory And I/O Cache Performance** – Reducing cache miss penalty and miss rate – Reducing hit time Main memory and performance – Memory technology. Types of storage devices – Buses – RAID – Reliability, availability and dependability – I/O performance measures designing an I/O system.

#### **Unit V**

**Multi-Core Architectures** Software and hardware multithreading – SMT and CMP architectures – Design issues – Case studies – Intel Multi-core architecture – SUN CMP architecture – heterogeneous multi-core processors – case study: IBM Cell Processor.

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

**Upon successful completion of this subject students should be able to:**

CO1: Know the fundamental aspects of computer architecture design and analysis

CO2: Understand design, pipelining, instruction set, out-of-order execution, caches (memory hierarchies).

CO4: Understand virtual memory, storage systems, and simulation techniques

CO3: Know about computer performance, instruction set architecture design and implementation.

CO5: Know about Multi-Core Architectures, Software and hardware multithreading

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books:**

1. “Advanced Computer Architecture for Parallel Processing (Mcgraw Hill Series in Electrical and Computer Engineering)” by Kai Hwang
2. “Advanced Computer Architecture for Parallel Processing”, Wiley, Hesham El-Rewini and Mostafa Abd-El-Barr

**Reference Books**

1. David E. Culler, Jaswinder Pal Singh, “Parallel computing architecture :Ahardware/software approach” , Morgan Kaufmann /Elsevier Publishers, 1999.
2. “Computer Architecture and Parallel Processing” by Bharat Bhushan Agarwal and Sumit Prakash Tayal.
3. “Parallel Computers – Architecture and Programming” by Rajaraman V



4. Kai Hwang and Zhi.WeiXu, “Scalable Parallel Computing”, Tata McGraw Hill, NewDelhi, 2003.

**Course Name: Theory of Computation**

**Course Code: MCA152A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

### **Course Objective**

1. To introduce the mathematical foundations of computation including automata theory.
2. To understand the concept of formal languages and grammars and the notions of algorithm, decidability, complexity, and computability.
3. To enhance/develop students' ability to understand and conduct mathematical proofs for computation and algorithms.
4. To enhance the ability to understand the concept of Regular Expression Formalism.
5. To be able to describe formation of Turing machine by Grammars.

### **Syllabus**

#### **Unit I**

**Introduction:** Basic Concepts, Sets, Relations, Graphs, Languages, Mathematical Finite State Machines FSM: Concept of Basic Machine Induction,, Finite State Machine, Finite Automata, Deterministic Finite Automata, Non-deterministic Finite Automata, Equivalence of NFA and DFA, NFA with  $\epsilon$ -Transitions, Equivalence of NFA and NFA with  $\epsilon$ -Transitions, Equivalence of DFA and NFA with  $\epsilon$ -Transitions, Finite Automata with Output, Equivalence of Moore and Mealy Machines, FSM Equivalence, DFA Minimization

#### **Unit II**

**Regular Expression:** Formalism, Examples of Regular Expressions, Equivalence of Regular Expressions and Finite Automata, DFA to Regular Expression Conversion, Regular Sets and their Closure Properties, Pumping Lemma for Regular Languages, Decision Algorithms for Regular Sets, Applications of Regular Expressions and Finite Automata.

### **Unit III**

**Turing Machine:** Elements of a Turing Machine, Turing Machine Formalism, Instantaneous Description, Transition Graph for Turing Machine, Complexity of a Turing Machine, Composite and Iterative Turing Machines, Universal Turing Machine, Multi-tape Turing Machine, Multi-stack Turing Machine, Multi-track Turing Machine, Solvable, Semi-solvable, and Unsolvable Problems, Halting Recursively Enumerable and Recursive Languages, Functions, Linear Problem Bounded Automata.

### **Unit IV**

**Grammar:** Constituents of Grammar, Formal Definition of Grammar, Grammar Notations, Derivation Process, Derivation Tree, Context-free Languages, Ambiguous Context-free Grammar, Simplification of Context-free Grammar, Normal Forms, Chomsky Hierarchy, Equivalence of Right-linear and Left-linear Grammars, Equivalence of Regular Grammars and Finite Automata Pumping Lemma for Context-free Languages, Kuroda Normal Form, Dyck Language, Derivation Graph, Applications of Context-free Grammar, Backus–Naur Form

### **Unit V**

**Automata:** Elements of a PDA, Pushdown Automata, Finite Automata vs PDAPDA Accepting CFLs, DPDA vs NPDA, Equivalence of CFG and PDA, Closure Properties of CFLs, Additional PDA Examples. Parsing, Bottom-up Parsing, Working of a Bottom-up Parser, Automatic Construction of Bottom-up Parser.

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

**On successful completion of this course, the learner will be able to**

CO1: Students are able to understand basic terms TOC and FSM.

CO2: Enhance the knowledge Regular Expression and its fundamentals.

CO3: Determine formation of Turing machine by Grammars.

CO4: Students will be able to understand the different-2 types of grammars

CO5: Students will be able to describe the PDA, Parsing, Bottom-up Parsing.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Vivek Kulkarni "Theory of Computation" , Oxford University Press. 2013
2. Peter Linz, "An Introduction to Formal Languages and Automata", (5e), Jones & Bartlett Learning, 2011.

**Reference Books:**

1. John C Martin, "Introduction to Languages and the Theory of Computation", (3e), McGraw Hill, 2007.
2. J E Hopcroft, Rajeev Motwani & Jeffrey D Ullman, "Introduction to Automata Theory, Languages and Computation", (3e), Pearson Education, 2006.
3. K. L. P. Mishra, N. Chandrashekharan, "Theory of Computer Science", (3e), PHI publications, 2007.

**Course Name: Data Warehousing and Data Mining**

**Course Code: MCA153A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objective:**

1. To know the basic concept of data mining, Architecture of a Data Mining System and the Knowledge Discovery Process
2. To provide the fundamental concepts of data warehousing technology.
3. To incorporate a step-by-step approach to designing and building a data warehouse.
4. To provides numerous review questions, multiple choice questions and other exercises at the end of each chapter.
5. To contain a running fabricate case-study to bring out practical aspects of building a data warehouse.

**Syllabus**

**Unit I**

**Introduction to Data mining:** Introduction, Architecture of a Data Mining System, The Knowledge Discovery Process , Integrating Data Mining with Data Warehouse, Related Areas of Data Mining , Data Mining Techniques

**Unit II**

**Data Warehousing:** Introduction to Data Warehousing: Historical Background, Increasing Demand for Strategic Information, Data Warehouse Defined, Data Warehouse Users, Benefits of

Data Warehousing, Concerns in Data Warehousing. Features of a Data Warehouse, Data Granularity, the Information Flow Mechanism, Metadata, Two Classes of Data, The Lifecycle of Data, Data Flow from Warehouse to Operational Systems.

### **Unit III**

**Architecture, Planning and Project Management:** Architecture, Characteristics of Data Warehouse Architecture, Data Warehouse Architecture Goals, Data Warehouse Architecture, Data Warehouse and Data Mart, Building Data Marts and its issues, Building Data Marts, Other Data Mart Issues.

Planning and Project Management: Project Management Principle, Data Warehouse Readiness Assessment, Data Warehouse Project Team, Planning for the Data Warehouse, Data Warehouse Project Plan, Planning for the Data Warehouse Server, Capacity Planning, Selecting the Operating System, Selecting the Database Software.

### **Unit IV**

**Schema & Dimensional Modeling:** Schema: Dimensional Modeling, The Star Schema, The Snowflake Schema, Aggregate Tables, Fact Constellation Schema, The Strengths of Dimensional Modeling, Data Warehouse and the Data Model.

Dimensional Modeling: Characteristics of a Dimension Table, Characteristics of a Fact Table, T Types of Dimension Tables, Keys in the Data Warehouse (Star) Schema, Enhancing the Data Warehouse Performance, Technology Requirements.

### **Unit V**

**Testing, Growth, Maintenance & OLAP:** Data Warehouse Design Review, Developing the Data Warehouse Iteratively, Testing, Monitoring the Data Warehouse, Tuning the Data Warehouse. OLAP: Need For OLAP, OLAP and Multidimensional Analysis, OLAP Functions, OLAP Applications, OLAP Models, OLAP Design Considerations, OLAP Tools and Products, Existing OLAP Tools, Data Design, Administration and Performance, OLAP Platforms.

**Course Outcomes (COs):**

**On successful completion of this course, the learner will be able to**

CO1: Describe the fundamental concepts, benefits and problem areas associated with data warehousing.

CO2: Describe the various architectures and main components of a data warehouse.

CO3: Design a data warehouse, and be able to address issues that arise when implementing a data warehouse.

CO4: Understand various schemes and data models.

CO5: Compare and contrast OLAP and data mining as techniques for extracting knowledge from a data warehouse.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books:**

1. Jiawei Han and MichelineKamber, “Data Mining- Concepts and Techniques”, (3e), Morgan Kaufmann Publishers, 2011
2. VikramPudi, "Data Mining" , Third Edition, Oxford university Press

**Reference Books :**

1. PaulrajPonniah, “Data Warehousing”, (2e), Wiley India Pvt. Ltd., 2010

2. H. Witten and E. Frank, “Data Mining: Practical Machine Learning Tools and Techniques”, Morgan Kaufmann, 2000.
3. M.H.Dunham, “Data Mining: Introductory and Advanced Topics”, Pearson Education 2006.

**Course Name: Optimization Techniques**

**Course Code: MCA154A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objective:**

1. To introduce the fundamental concepts of Optimization Techniques.
2. To make the learners aware of the importance of optimizations in real scenarios.
3. To provide the concepts of various classical and modern methods of for constrained and unconstrained problems in both single and multivariable.
4. To introduce the concept of modeling approach.
5. To introduce the concept of linear programming.

**Syllabus**

**Unit I**

**Nature of Operation Research:** History, Nature of OR, Impact of OR, Application Areas. Overview of modeling approach: Formulating the problem, constructing a mathematical model, Deriving a solution, Testing a model and the solution, establishing control over the solution, Implementation issues.

**Unit II**

**Linear Programming:** Introduction, Graphical solution, Graphical sensitivity analysis, The standard form of linear programming problems, Basic feasible solutions, Simplex algorithm, Artificial variables, Big M and two phase method, Degeneracy, Alternative optima, Unbounded solutions, Infeasible solutions.

### Unit III

**Dual Problem:** Relation between primal and dual problems, Dual simplex method. Assignment problem: Hungarian method.

### Unit IV

**Travelling Salesman Problem:** Branch and Bound Technique. Sequencing Problem: 2 machines n jobs, 3 machines n jobs, n machines m job.

### Unit V

**Pert and CPM:** Arrow network, Time estimates, earliest expected time, latest allowable occurrence time, latest allowable occurrence time and slack, Critical path, Probability of meeting scheduled date of completion of project, Calculation of CPM network, Various floats for activities, Project crashing.

### Course Outcomes (COs):

CO1: Formulate optimization problems.

CO2: Understand and apply the concept of optimality criteria for various type of optimization problems.

CO3: Solve various constrained and unconstrained problems in single variable as well as multivariable.

CO4: Apply the methods of optimization in real life situation.

CO5: Know about the Calculation of CPM network and Pert.

### MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H		H					H	M			
CO2				H		H				L		



CO3												M
CO4				M				H				
CO5		H					M		H		M	

H = Highly Related; M = Medium; L = Low

**Text Books:**

1. Gillet, B.E., "Introduction to Operation Research: a computer oriented algorithmic approach" Tata McGraw Hill, NY.
2. KantiSwaroop, Gupta P.K. Man Mohan, "Operations Research", Sultan Chand and Sons.

**References Books:**

1. Kambo, N.S., "Mathematical Programming Techniques", McGraw Hill
2. Operations Research : Principles and Practice 2nd edition Ravindran Wiley Production.
3. Elementary Number Theory with applications: Thomas Koshy, 2nd Ed., Elsevier.

**Course Name: NLP and Information Retrieval**

**Course Code: MCA155A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objective:**

1. To introduces the fundamental concepts and techniques of natural language processing (NLP).
2. To gain an in-depth understanding of the computational properties of natural languages and the commonly used algorithms for processing linguistic information.
3. To examines NLP models and algorithms using both the traditional symbolic and the more recent statistical approaches.
4. To understanding semantics and pragmatics of English language for processing .
5. To be able to write programs in Python to carry out natural language processing

**Syllabus**

**Unit I**

**Introduction to NLP:** What is Natural Language Processing (NLP), Origins of NLP, Language and Knowledge, The Challenges of NLP, Language and Grammar, Processing Indian Languages, NLP Applications, Some Successful Early NLP Systems, Information Retrieval Language Modeling: Various Grammar-based Language Models, Statistical Language Model

**Unit II**

**Word Level Analysis:** Regular Expressions, Finite-State Automata, Morphological Parsing, Spelling Error Detection and Correction, Words and Word Classes, Part-of-Speech Tagging, Syntactic Analysis: Context-Free Grammar, LEEE Contents, Constituency, Parsing, Probabilistic Parsing, Indian Languages

### **Unit III**

**Semantic Analysis:** Meaning Representation, Lexical Semantics, Ambiguity, Word Sense Disambiguation. Discourse Processing: Introduction, Cohesion, Reference Resolution, Discourse Coherence and Structure

### **Unit IV**

**Natural Language Generation & Machine Translation:** Architectures of NLG Systems, Generation Tasks and Representations, Applications of NLG. Machine Translation: Introduction, Problems in Machine Translation, Characteristics of Indian Languages, Machine Translation Approaches, Direct Machine Translation, Rule-based Machine Translation, Corpus-based Machine Translation, Semantic or Knowledge-based MT systems, Translation involving Indian Languages

### **Unit V**

**Information Retrieval:** Basics, Design Features of Information Retrieval systems, Contents EN, Information Retrieval Models, Classical Information Retrieval Models, Non-classical models of IR, Alternative Models of IR, Evaluation of the IR System

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

**Upon successful completion of this subject students will able**

CO1: The students will get acquainted with natural language processing and learn how to apply basic algorithms in this field.

CO2: They will understand the algorithms of Word level analysis.

CO3: They will understand the algorithms of Semantic analysis.

CO4: They will also grasp basics of knowledge representation Natural languages and machine Translation.

CO5: The students will get acquainted with the methods of information retrievals.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books:**

1. Natural Language Processing and Information, Oxford University, Siddiqui, Tanveer and Tiwary, April 2008.
2. “Foundations of Statistical Natural Language Processing” by Chris Manning and Hinrich Schuetze.

**Reference Books**

1. “Natural Language Processing Tools” by Elaine UíDhonnchadha and Elaine U. Dhonnchadha.
2. “Natural Language Processing Using Very Large Corpora” by S Armstrong and Kenneth W Church.
3. “Speech and Language Engineering HRD Edition” by Rajman Pallota.

**Course Name: Object Oriented Analysis and Design**  
**Course Code: MCA156A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objective:**

1. To teach the student the essential and fundamental aspects of object oriented analysis and design, in terms of “how to use” it for the purpose of specifying and developing software.
2. To explore and analyze different analysis and design models, such OO Models, Structured Analysis and Design Models, etc.
3. To understanding the insight and knowledge into analyzing and designing software using different object-oriented modeling techniques.
4. To understanding the fundamental principles through advanced concepts of analysis and design using UML
5. To providing clear instructions and information on the "How-to" dimension for applying the UML models and to the ways to document their products

**Syllabus**

**Unit I**

**Introduction to object oriented systems:** Classes, Objects, Abstraction, Inheritance, Polymorphism, Encapsulation, Message Sending, Association, Aggregation, Iterative development and the Unified Process (UP), UP phases: Inception, Elaboration, Construction and Transition, Object-oriented metrics

**Unit II**

**Introduction to UML:** Use Cases and functional requirements, Identifying and writing Use Cases, Decomposition of use cases, Modeling System Workflows using Activity Diagrams, Modeling a System's Logical Structure using Classes and Class Diagrams, Modeling Interactions using Sequence Diagrams and Communication Diagrams, Timing Diagrams, Interaction

Overview Diagrams, Component Diagram, Package diagram, State Machine Diagrams, Deployment Diagrams.

### **Unit III**

**Introduction to Patterns:** GoF Patterns, Creational Patterns, Structural Patterns, Behavioral Patterns, Software Architectural patterns, The Observer Pattern, The Template Method Pattern , Factory Patterns: Factory Method and Abstract Factory.

### **Unit IV**

**Pattern:** The Singleton Pattern, The Iterator Pattern, The Composite Pattern, The Facade Pattern , The State and Strategy patterns , Command Pattern , The Adapter Pattern , The Proxy Pattern , The Decorator Pattern, The Visitor Pattern , Anti Patterns, Patterns for Assigning Responsibilities: GRASP Patterns

### **Unit V**

**Domain modeling:** assigning responsibility using sequence diagrams, mapping design to code, CASE tools, Unit, Cluster, and System-level testing of Object-oriented programs, Aspect-oriented and Service-oriented software.

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

**Upon successful completion of this subject students should be able to:**

CO1: Explain the object oriented software development process, including object-oriented methodologies and work flows.

CO2: Justify designs based on design principles, patterns, and heuristics

CO3: Know about Domain modeling, System-level testing of Object-oriented programs, Aspect-oriented and Service-oriented software.

CO4: Create use case documents that capture requirements for a software system.

CO5: Create class diagrams that model both the domain model and design model of a software system.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Applying UML and Patterns: An Introduction to object-oriented Analysis and Design and iterative development, by Craig Larman, Pearson Education. (1998)
2. Design Patterns - Elements of Reusable Object-Oriented Software, Gamma, et. al., Addison-Wesley. (1994)

**Reference Books:**

1. BjarneStroustrup, The C++ Programming Language, Addison-Wesley.
2. Andrei Alexandrescu, Modern C++ Design, Pearson Education India.
3. Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software, Addison Wesley.

**Course Name: Cloud Computing**

**Course Code: MCA157A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objective:**

1. To provide students with the fundamentals, essentials of Cloud Computing and cloud models.
2. To be able to work with cloud services and to provide a sound foundation of the Cloud Computing so that they are able to start using and adopting Cloud Computing services and tools in their real life scenarios.
3. To learn about the cloud environment, building software systems and components that scale to millions of users in modern internet.
4. To understand basic and advance services provide by the cloud and basic architecture on which cloud is based upon.
5. To enable students exploring some important cloud computing driven commercial systems such as Google Apps, Microsoft Azure and Amazon Web Services and other businesses cloud applications.

**Syllabus**

**Unit I**

**Introduction:** Business and IT perspective, Cloud and virtualization, Cloud services requirements, cloud and dynamic infrastructure, cloud computing characteristics, cloud adoption. Cloud models: Cloud characteristics, Measured Service, Cloud models, security in a public cloud, public verses private clouds, cloud infrastructure self service.

**Unit II**

**Cloud at a service:** Gamut of cloud solutions, principal technologies, cloud strategy, cloud design and implementation using SOA, Conceptual cloud model, cloud service demand.



### **Unit III**

**Cloud solutions:** Cloud ecosystem, cloud business process management, cloud service management, cloud stack, computing on demand, cloud sourcing.

### **Unit IV**

**Cloud management:** Resiliency, Provisioning, Asset management, cloud governance, high availability and disaster recovery, charging models, usage reporting, billing and metering.

### **Unit V**

**Cloud and SOA:** SOA journey to infrastructure, SOA and cloud, SOA defined, SOA defined, SOA and IAAS, SOA based cloud infrastructure steps, SOA business and IT services.

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

**Upon successful completion of this subject students should be able to:**

CO1: To provide students with the fundamentals, essentials of Cloud Computing and cloud models.

CO2: Students are able to work with cloud services and to provide a sound foundation of the Cloud Computing so that they are able to start using and adopting Cloud Computing services and tools in their real life scenarios.

CO3: Understand about the cloud solutions, Cloud ecosystem, cloud business process management.

CO4: To learn about designing and implement cloud computing application and cloud management.

CO5: Demonstrate the SOA journey to infrastructure, SOA and cloud and IT services.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books:**

1. Cloud Computing by Dr. Kumar Saurabh, Wiley India, 2011.
2. Michael Miller, Cloud Computing: Web based applications that change the way you work and collaborate online, Que publishing, August 2009.

**Reference Books**

1. RajkumarBuyya, James Broberg, Andrzej M. Goscinski,"Cloud Computing:Principles and Paradigms", Edition1, Wiley,2011
2. Barrie Sosinsky,"Cloud Computing Bible",Edition1, Wiley-India, 2010
3. Ronald L. Krutz, Russell Dean Vines,"Cloud Security:A Comprehensive Guide to Secure Cloud Computing", Edition 1 , Wiley- India,2010

**Course Name: Distributed Computing**

**Course Code: MCA158A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objective:**

1. To be familiar with the differences among: concurrent, networked, distributed, and mobile.
2. To be familiar with the Resource allocation, Inter Process and remote Communication.
3. To be familiar with the deadlock detection and avoidance techniques in distributed system.
4. To gain the knowledge of concept of shared memory in Distributed environment.
5. To be familiar with the Distributed file system, it's Implementation and new trends in distributed file systems and their naming.

**Syllabus**

**Unit I**

**Introduction:** Distributed System definition , Architectures for distributed systems, Distributed Computing Models , Software concepts , Network Operating System , Distributed Operating System , Multiprocessor Time-sharing System , Comparison of operating systems , Issues in designing Distributed Systems , Client–Server Model , Case Studies

Network Communication: LAN and WAN Technologies , Introduction to LAN and WAN , Classification of networks , Protocols for Network Systems , The ISO/OSI Reference Model , Internet Protocols , Asynchronous Transfer Mode , Protocols for distributed systems

**Unit II**

**Inter-process and Remote Communication:** Inter-process Communication: Message Passing, Introduction to Message Passing, Advantages and Features of Message- passing Systems , IPC Message format, Group Communication,

Remote Communication: Introduction to remote communication, Middleware, Remote Procedural Call Basics, RPC Implementation, RPC Communication

### **Unit III**

**Synchronization:** Synchronization in distributed systems, Clock synchronization Mutual exclusion , Election algorithms , Atomic transactions , Deadlocks in distributed system , Threads, Thread usage and Implementation of thread packages, processor allocation

### **Unit IV**

**Distributed Shared Memory:** Basic concepts of DSM, DSM architecture, Message passing vs shared memory, Types of DSMs, Advantages of DSM, Hardware DSM, On-chip memory DSM, Bus-based multiprocessor, Ring-based multiprocessor, Design Issues in DSM Systems , Granularity, Consistency models, Coherence protocols, Issues in implementing DSM Systems, Thrashing, Responsibility for DSM management, Replication vs migration strategies ,Replacement Strategy ,Heterogeneous and Other DSM Systems

### **Unit V**

**File Systems and Naming:** Distributed File system: File service interface, semantics of the file sharing, Distributed file system, Implementation of new trends in distributed file systems  
Naming: Features of Good Naming, System Oriented Names, Object, Locating Mechanisms, Name Caches, Naming and Security

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

**Upon successful completion of this subject students will able**

CO1: Describe the architecture and model of distributed computing and communication over network.

CO2: Develop, test and debug RPC based client-server programs in Unix.

CO3: Design and build application programs synchronous distributed systems.

CO4: Improve the performance and reliability of distributed shared memory.

CO5: Design and build newer distributed file systems for any OS.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books:**

1. Distributed Computing, Oxford University, Mahajan and Shah, June 2013.
2. “Distributed Computing South Asian Edition: Principles, Algorithms, and Systems” by Professor Ajay D Kshemkalyani and Professor MukeshSinghal.

**Reference Books**

1. “Distributed and Cloud Computing: From Parallel Processing to the Internet of Things” by Hwang
2. “Distributed Computing: Principles and Applications” by Liu
3. “Distributed Computing” by Sunita Mahajan and Seema Shah

**Course Name: Introduction to Sales Force**

**Course Code: MCA159A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objective:**

1. To introduce cloud computing and sales force approaches.
2. To understand the concept and features of CRM, Data Migration Import Wizard and Data Loader.
3. To describe needs of APEX as a Programming Language, Various Statements and Operators.
4. To demonstrate class and visual force.
5. To be able to understand the concept of Trigger Context Variables, Validation and Automation, Components, Client Controller and Server Controller.

**Syllabus**

**Unit I**

**Introduction to Cloud Computing & Sales force:** On premise and Cloud approach, IaaS, PaaS and SaaS, Clouds, Versions, Editions & Environments, Product & Project Development MVC Pattern and Multitenant Architecture: Configuration, Development, Testing and Admin, along with 15 steps of Salesforce.com, MVC Diagram, Multi-Tenant Diagram  
Starting Designing Applications on Force.com: Basic Vocabulary, Registration, Navigation and Principles, Understanding the Same with Lightning, Relationships, All 6 Types, Validation, Page Layouts and Record Types

**Unit II**

**CRM:** Campaign, Lead, Account, Contact and Opportunity, Campaign, Lead, Account, Contact and Opportunity, Product, Process Book, Quote, Forecast, Order, Asset, Case, Solution, Activity, Task, Event, Calendar, PaaS features for CRM.

Data Migration Import Wizard and Data Loader: Reports and Dash-Boards, Work flows, Process Builder, Approval Process, Security Profile, Role, Queues, Public Group, OWD, Sharing Rules, Permission Sets.

### **Unit III**

**Introduction to Programming:** Need of APEX as a Programming Language, Various Statements and Operators, 9 types of operators and 5 types of statements, System.debug, Comments, Indentation, Naming Conventions, OOP, Business Class , Test Class, Test Methods, Creating object, calling methods, executing a program, Constructor / Method / Operator Over Loading / this, Default , Parameterised .

### **Unit IV**

**Class and Visual Force:** Static and non-static members of a class, Class Summary, try catch, Collections list: Set, Map, Exception Handling, Assertions and Annotations: 6 types of Assertions and 3 Annotations, Database Statements: DML , MC inter relation, Database Statements : SOQL / SOSL, Visual Force - Types of Tags: 6 Types, Visual Force -Static / Dynamic Data, Org Subject \$User: {! } \$User, Standard Controllers: Save, Save and New, Cancel ,Edit, Delete, Explanation of for each loop / Standard list Controller : Previous, Next, First Last, Custom Controller, VC / MVC Examples: Add numbers, Insert and fetch Record

### **Unit V**

**Custom List Controller:** Insert and fetch Records, Schedule Apex: Interface, Batch Apex: 50 K, 50 Million, Triggers: if and its types, Trigger Context Variables, Validation and Automation, Components, Client Controller, Server Controller

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

**Upon successful completion of this subject students will able**

CO1: Introduce cloud computing and sales force approaches.

CO2: Able to understand the concept and features of CRM, Data Migration Import Wizard and Data Loader.

CO3: Describe needs of APEX as a Programming Language, Various Statements and Operators.

CO4: To demonstrate class and visual force.

CO5: Able to understand the concept of Trigger Context Variables, Validation and Automation, Components, Client Controller and Server Controller.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books:**

1. Wes Nolte , Jeff Douglas , “Salesforce Handbook” (3e), Lulu.com, 2011.
2. David Taber, “Salesforce.com Secrets of Success: Best Practices for Growth and Profitability”, 1 edition, Prentice Hall, 2009.

**Reference Books:**

1. Tom Wong, Liz Kao & Matt Kaufman “Salesforce for Dummies”,4th Revised edition edition ,John Wiley & Sons, 2010
2. Visual force practices by Michael Floyd, Don Robins, Dan Appleman, et al.
3. Development with the Force.com Platform, Jason Ouellette



**Course Name: Design & Analysis of Algorithms**  
**Course Code: MCA166A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objective:**

1. To design effective, efficient, elegant, and readable algorithms for various classes of computing problems
2. To learn good principles of algorithm design;
3. To determine space and time complexities of algorithms by the use various algorithm design techniques like (divide and conquer, backtracking, greedy, etc.)
4. To become familiar with fundamental data structures and with the manner in which these data structures can best be implemented; become accustomed to the description of algorithms in both functional and procedural styles;
5. To learn how to apply their theoretical knowledge in practice

**Syllabus**

**Unit I**

Introduction, algorithms specification, time and space complexity, performance analysis, recurrence relations. Divide and Conquer – finding max min.

**Unit II**

Dynamic Programming and Greedy Methods – Huffman tree construction, Knapsack problem, 0/1 Knapsack problem, least common subsequence, matrix chain multiplication. Backtrack: 4-queen problem, Branch and Bound: assignment problem

**Unit III**

Graph algorithms–flow problems, String Matching Algorithms: Naive algorithm, automata and KMP matcher algorithms, Boyer-Moore algorithm

**Unit IV**

Number Theory Problems – CRT, GCD algorithms, modular arithmetic, Lower Bound Theory;  
Approximate Algorithms – Set cover, vertex cover, .Randomized Algorithms – Las Vegas and Monte Carlo methods

## Unit V

NP Completeness: Definitions of P, NP-Hard and NP-Complete Problems. Decision Problems..

### Course Outcomes (COs):

**Upon successful completion of this subject students should be able to:**

CO1: Able to design effective, efficient, elegant, and readable algorithms.

CO2: What are the various methods of calculating complexity?

CO3: Which method is best for different algorithms?

CO4: About computational geometry and lower bound theory etc.

CO5: Able to solve the NP completeness problem.

### MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

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

H = Highly Related; M = Medium; L = Low

### Textbooks

1. Cormen, Leizerson&Rivest, Introduction to algorithms, Prentice-Hall. 2002
2. Horowitz &Sahni, Fundamentals of Computer Algorithms, Galgotia Publication. 1999

**Reference Books:**

1. Aho, Hopcroft, Ullman, The Design and Analysis of Computer Algorithms, Addison-Wesley. 2001.
2. Introduction to Design and Analysis of Algorithms, Anny Levitin, Person Education Press. 2007.
3. Gilles Brassard & Paul Bratley, Fundamental Algorithms, Prentice-Hall. 1998

**Course Name: Mobile Computing**

**Course Code: MCA167A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objective:**

1. To be able to understand the concepts of mobile computing and mobile communication in equal measure 2G, 2.5G, 3G, 3G+, and 4G communication systems.
2. To know about mobile satellite communication networks, mobile IP, mobile TCP and digital audio-video broadcasting
3. To be able to understand Mobile systems and devices, mobile operating systems used for application development, mobile databases,
4. To know about Client-server computing agents, application servers.
5. To know about security protocols, mobile Internet, and ad-hoc and sensor networks.

**Syllabus**

**Unit I**

**Mobile Communication and Mobile Devices:** Mobile Computing introduction, Mobile Computing, Mobile Computing Architecture, Mobile Devices, Mobile System, Networks, Data Dissemination, Mobility Management, Security.

Cellular Networks and Frequency Reuse, Mobile Smartphones, Smart Mobiles, and Systems, Handheld Pocket Computers, Smart Systems, Limitations of Mobile Devices, Automotive Systems.

**Unit II**

**GSM and Other 2G Architectures :** GSM—Services and System Architecture, Radio Interfaces of GSM, Protocols of GSM, Localization, Call Handling, Handover, Security, New data Services, General Radio Packet Services.

### **Unit III**

**Wireless Medium Access Control, CDMA, 3G and 4G Communication:** Modulation, Multiplexing, Controlling the Medium Access, Spread Spectrum, FHSS, Coding Methods, CDMA, OFDM, High Speed Packet Access (HSPA) 3G Network, Long-term Evolution, WiMaxRel 1.0 IEEE 802.16e, Broadband Wireless Access, 4G Networks, Mobile Satellite Communication Networks.

### **Unit IV**

**Mobile IP Network Layer and Mobile Transport Layer:** IP and Mobile IP Network Layers, Packet Delivery and Handover Management, Location Management, Registration, Tunnelling and Encapsulation, Route Optimization, DHCP, VoIP, Ipsec. Conventional TCP/IP Transport Layer Protocols, Indirect TCP, Snooping TCP, Mobile TCP, Other Methods of Mobile TCP-layer Transmission, TCP over 2.5G/3G Mobile Networks.

### **Unit V**

**Mobile Ad-hoc and Wireless Sensor Networks:** Introduction to Mobile Ad-hoc Network, Fixed Infrastructure Architecture, MANET Infrastructure Architecture, MANET, Properties, Spectrum; Applications Routing and Routing Algorithms, Security in Ad-hoc Network, Wireless Sensor Networks, Data Dissemination, Sensor Network Applications, Distributed Network and Characteristics, TinyOS.

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

**Upon successful completion of this subject students should be able to:**

CO1: Describe wireless and mobile communications systems and be able to choose an appropriate mobile system from a set of requirements.

CO2: To describe all layers, GSM and Architectures

CO3: To describe Interface a mobile computing system with CDMA, 3G and 4G communications.

CO4: To describe IP Network and layers.

CO5: To describe MANET and wireless sensor networks

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Mobile Computing, Oxford University, Raj Kamal, December 2011.
2. UweHansmann, LotharMerk, Martin S. Nicklons and Thomas Stober-  
“Principles of Mobile Computing”

**Reference Books**

1. Mobile Communication 2nd edition by Jochen Schiller, Pearson education
2. Mobile Computing by AsokeTalukder, RoopaYavagal (Tata McGraw Hill)
3. HazysztofWesolowshi- “Mobile Communication Systems”

**Course Name: Image Processing**

**Course Code: MCA168A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objective:**

1. To introduce students to the mathematical foundation of Image processing
2. To do detailed study of image processing Operations and Systems.
3. To learn students about Image transforms, Enhancement with algorithms of restoration and Compression.
4. *To do detailed study about image segmentation.*
5. To learn about Image Compression Model and Compression Algorithm.

**Syllabus**

**Unit I**

**Introduction to Image Processing :** Overview of Image Processing, Nature of Image Processing, Image Processing and Related Fields, Digital Image Representation Types of Images, Digital Image Processing Operations, Fundamental Steps in Image Processing, Image Processing Applications.

**Unit II**

**Digital Imaging Systems and Processing Operations:** Overview of Digital Imaging Systems, Physical Aspects of Image Acquisition, Biological Aspects of Image Acquisition, Review of Digital Cameras, Image Display Devices and Device Resolution, Digital Halftone Process, Image Storage and File Formats. Basic Relationships and Distance Metrics, Classification of Image Processing Operations, Logical Operations, Geometrical Operations, Image Interpolation Techniques.

### **Unit III**

**Image Enhancement:** Image Quality and Need for Image Enhancement, Image Enhancement Operations, Image Enhancement in Spatial Domain , Histogram-based Techniques, Spatial Filtering Concepts, Image Smoothing Spatial Filters, Image Sharpening Spatial Filters, Image Smoothing in Frequency Domain Filtering, Image Sharpening in Frequency Domain, Homomorphic Filtering.

### **Unit IV**

**Image Restoration and Compression:** Image Restoration : Introduction to Degradation, Types of Image Degradations, Point Spread Function and Modulation Transfer Function, Image Degradation Model, Noise Modelling, Image Restoration in Presence of Noise Only, Periodic Noise, and Band-pass and Band-reject Filtering, Image Restoration Techniques.

**Image Compression:** Image Compression Model, Compression Algorithm and its Types, Types of Redundancy, Lossless Compression Algorithms, Lossy Compression Algorithms, Image and Video Compression, Wavelet Transform and Multiresolution Analysis.

### **Unit V**

**Image Segmentation and Colour Image Processing: Segmentation:** Introduction, Classification of Image Segmentation Algorithms, Detection of Discontinuities, Edge Detection, Hough Transforms and Shape Detection, Corner Detection, Principle of Thresholding , Principle of Region-growing, Active Contour Models, Validation of Segmentation Algorithms.

**Colour Image Processing:** Colour Image Storage and Processing, Colour Models, Colour Quantization, Pseudocolour Image Processing, Full Colour Processing.

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

**On successful completion of this course, the learner will be able to :**

CO1: Students are able to understand basic terms of Image Processing.

CO2: Students are able to understand Processing operations and its fundamentals.

CO3: Students will be able to describe the effect of transform operation on Digital Image .

CO4: Students will be able to understand the different-2 Image compressing algorithms.

CO5: Students will be able to describe segmentation and Color image processing.



**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Rafael C. Gonzalez, Richard E. Woods, “Digital Image Processing” (3e), Pearson Prentice Hall, 2008.
2. S. Sridhar , "Digital Image processing" , Oxford University Press, 2011

**Reference Books**

1. Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins, “Digital Image Processing Using MATLAB”, (2e), Tata McGraw Hill Publication, 2010.
2. Sonka, Hlavac, Boyle, “Digital Image Processing and Computer Vision”, CENGAGE Learning.
3. “Digital Image Processing” by Rafael C Gonzalez and Richard E Woods.

**Course Name: Soft Computing**

**Course Code: MCA169A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objective:**

1. To illustrative description of the inherent principles of soft computing, eliminating heavy mathematical treatment
2. To introduce the concept of swarm intelligent systems
3. To extensive the coverage of neural networks and fuzzy logic concepts
4. To understanding basic concepts and working of generic algorithms.
5. To understand the concept of artificial neural network.

**Syllabus**

**Unit I**

**Introduction to soft computing, Need for soft computing,** Artificial Intelligence, Artificial Neural Networks , Fuzzy Systems, Genetic Algorithm and Evolutionary Programming, Swarm Intelligent Systems, Swarm Intelligent Systems, Comparison among Intelligent Systems

**Unit II**

**Artificial Neural Networks- I:** Introduction to Neural Networks, Biological Inspiration, Comparison between Brain and Computer, Biological Neural Networks to Artificial Neural Networks, Information Processing at the Neurons and Synapses, Classification of ANNs, Neural Network Architecture, Learning/Training, Training and Testing Modes, Activation/Transfer Function, First-generation Neural Networks, McCulloch and Pitts Neuron Model, Learning Rules: Hebbian and Delta, Perceptron Network , Adaline Network , Madaline Network

**Unit III**

**Artificial Neural Networks- II:** Backpropagation Neural Networks, Kohonen Neural Network, Learning Vector Quantization, Hamming Neural Network, Hopfield Neural Network, Bi-

directional Associative Memory, ART Network, Boltzman Machine Neural Networks , Radial Basis Function Neural Networks, Support Vector Machines.

#### **Unit IV**

**Fuzzy Logic:** Fuzzy Logic Introduction, Human Learning Ability, Undecidability, Imprecision, and Uncertainty, Probability Theory vs Possibility Theory, Fuzzy sets and Classical sets, Fuzzy set operations, Fuzzy relations, Fuzzy compositions, Natural Language and Fuzzy Interpretations, Structure of Fuzzy Inference System

#### **Unit V**

**Genetic Algorithm:** Basic concepts, working principle, procedures of GA, flow chart of GA, Genetic representations, (encoding) Initialization and selection, Genetic operators, Mutation, Generational Cycle, applications

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

**On successful completion of this course, the learner will be able to :**

CO1: To know about the basics of soft computing techniques and also their use in some real life situations.

CO2: To solve the problems using neural networks techniques.

CO3: To find the solution using different fuzzy logic techniques

CO4: To use the genetic algorithms for different modelling

CO5: To integrate the various soft computing techniques

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books**

1. “Soft Computing and Intelligent Systems Design: Theory, Tools and Applications” by Karray
2. Soft Computing with MATLAB Programming, Oxford University, Padhy and Simon, May 2015.

**Reference Books**

1. “Soft Computing :Fundamentals And Applications” by Pratihari D K
2. “Soft Computing Applications and Intelligent Systems” by ShahrulAzman Noah and SitiNorul Huda Sheikh Abdullah
3. “Soft Computing and Its Applications, Volume Two: Fuzzy Reasoning and Fuzzy Control: Volume 2” by Kumar S Ray

**Course Name: Compiler Design**

**Course Code: MCA170A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objectives:**

1. To understand, design and implement a lexical analyzer.
2. To understand, design and implement a parser.
3. To understand, design code generation schemes.
4. To understand optimization of codes and run time environment.
5. To understand various compiler writing tools.

**Syllabus**

**Unit I**

**Introduction to Compiler:** Theory of Computer Languages, Design of a Language, Evolution of Compilers, Stages of Compilation: Lexical Analysis, Syntactic Analysis, Semantic Analysis, Intermediate Code Generation, Code Optimization, Code Generation, Symbol Table Management, Error Management.

**Unit II**

**Lexical Analysis and Syntax Analysis:** Lexical Analysis: Alphabets and Tokens in Computer Languages, Representation of Tokens and Regular Expression, Token Recognition and Finite State Automata, Implementation, Error Recovery. Syntax Analysis: Context-free Grammar and Structure of Language, Parser and its Types, Top down Parser, Bottom-up Parser, Parser Generator Tool (Yacc).

**Unit III**

**Run-time Storage Organization Intermediate Code Generation:** Need for Intermediate Code, Types of Intermediate Code, Representations of All Language Constructs by Three-address,

Grammar Symbols and Attributes, Semantic Analysis, Semantic Routines for Intermediate Code Generation.

#### **Unit IV**

**Code Optimization:** Need for Optimization, Objectives, Performance factors, Writing Optimized Code at User Level, Construction of Basic Blocks and Processing, Data-flow Analysis Using Flow Graph, Principal Sources of Optimization and Transformations, Alias, Procedural Optimization, Loops in Flow Graphs, Loop Optimization.

#### **Unit V**

**Code Generations and Compiler Writing Tools:** Issues in Code Generation, Target Machine Architecture, Subsequent Use Information, Register Allocation, Directed Acyclic Graph Representation of Basic Blocks, Code Generation From Intermediate Code, Peephole Optimization, Code Scheduling, Lexical Tools, Syntactic Tools.

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

**On successful completion of this course, the learner will be able to :**

CO1: Learn the design and develop a comprehensive Compiler for a given language or an Introduction.

CO2: Analyze and able to design a compiler development with lexical and syntax analysis.

CO3: Describe all aspects of Run time storage with intermediate Code generation.

CO4: Implement various code optimization techniques.

CO5: Implement various optimization and code generation algorithms for the design of a compiler.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books**

1. “Compilers: Principles, Techniques and Tools” by Alfred V Aho and Ravi Sethi
2. Compiler Design, Oxford University, Muneeswaran, November 2012.

**Reference Books**

1. “Engineering a Compiler” by Keith D Cooper and Linda Torczon
2. “Compiler Design in C” by Allen I Holob
3. “Elements of Compiler Design” by Meduna

**Course Name: Big Data Analytics**

**Course Code: MCA171A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objective:**

1. To know the fundamental concepts of big data and analytics.
2. To explore tools and practices for working with big data
3. To learn about stream computing.
4. To know about the research that requires the integration of large amounts of data.
5. To know about clustering and classification.

**Syllabus**

**Unit I**

**Introduction To Big Data:** Evolution of Big data - Best Practices for Big data Analytics - Big data characteristics - Validating- The Promotion of the Value of Big Data - Big Data Use Cases- Characteristics of Big Data Applications - Perception and Quantification of Value - Understanding Big Data Storage – A General Overview of High-Performance Architecture - HDFS – Map Reduce and YARN – Map Reduce Programming Model

**Unit II**

**Clustering and Classification:** Advanced Analytical Theory and Methods: Overview of Clustering - K-means - Use Cases -Overview of the Method - Determining the Number of Clusters - Diagnostics - Reasons to Choose and Cautions .- Classification: Decision Trees - Overview of a Decision Tree – The General Algorithm - Decision Tree Algorithms - Evaluating a Decision Tree - Decision Trees in R - Naïve Bayes - Bayes' Theorem - Naïve Bayes Classifier.



### **Unit III**

**Association and Recommendation System:** Advanced Analytical Theory and Methods: Association Rules - Overview - Apriori Algorithm Evaluation of Candidate Rules - Applications of Association Rules - Finding Association & finding similarity - Recommendation System: Collaborative Recommendation- Content Based Recommendation - Knowledge Based Recommendation- Hybrid Recommendation Approaches.

### **Unit IV**

**Graph Memory And Stream Memory:** Using Graph Analytics for Big Data: Graph Analytics - The Graph Model - Representation as Triples - Graphs and Network Organization - Choosing Graph Analytics - Graph Analytics Use Cases - Graph Analytics Algorithms and Solution Approaches - Technical Complexity of Analyzing Graphs- Features of a Graph Analytics Platform - Considerations: Dedicated Appliances for Graph - 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 – Real time Analytics Platform(RTAP) applications - Case Studies - Real Time Sentiment Analysis, Stock Market Predictions.

### **Unit V**

**NoSQL Data Management For Big Data And Visualization:** NoSQL Databases : Schema-less Models: Increasing Flexibility for Data Manipulation-Key Value Stores- Document Stores - Tabular Stores - Object Data Stores - Graph Databases Hive -Sharding — Hbase – Analyzing big data with twitter - Big data for E-Commerce Big data for blogs- Review of Basic Data Analytic Methods using R.

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

**On successful completion of this course, the learner will be able to :**

CO1: Identify the difference between structured, semi-structured and unstructured data.

CO2: summarize the challenges of big data and how to deal with the same.

CO3: Explain the significance of NoSQL databases.

CO4: Explain about Hadoop Ecosystem.

CO5: Identify the difference between Pig and Hive.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Anand Rajaraman and Jeffrey David Ullman, "Mining of Massive Datasets", Cambridge University Press, 2012.
2. David Loshin, "Big Data Analytics: From Strategic Planning to Enterprise Integration

**Reference Books**

1. Tom White, "Hadoop: The Definitive Guide", Second Edition, O'Reilly Yahoo Press.
2. Robert D. Schneider, "Hadoop for Dummies", Wiley.
3. VigneshPrajapati, "Big Data Analytics with R and Hadoop", Packt Publishing.
- 4.

**Course Name: Organization Behavior**

**Course Code: MCA172A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objectives:**

1. To discuss the development of the field of organizational behavior and explain the micro and macro approaches
2. To analyze and compare different models used to explain individual behavior related to motivation and rewards
3. To identify the processes used in developing communication and resolving conflicts  
To explain group dynamics and demonstrate skills required for working in groups (team building)
4. To discuss the implementation of organizational change.

**Syllabus**

**Unit I**

**Organization behavior:** The Importance of Interpersonal Skills, Management Functions, Management Roles, Management Skills, Effective versus Successful Managerial Activities, Enter Organizational Behavior, Complementing Intuition with Systematic Study, Disciplines That Contribute to the Organizational Behavior Field, Psychology, Social Psychology, Sociology, Anthropology, Challenges and Opportunities for Organizational Behavior

**Unit II**

**Diversity in organization:** Demographic Characteristics of the U.S. Workforce, Levels of Diversity, Discrimination, Biographical Characteristics, Ability: Intellectual Abilities, Physical Abilities 55, The Role of Disabilities, Implementing Diversity Management Strategies, Attracting, Selecting, Developing, and Retaining Diverse Employees, Effective Diversity Programs.

### **Unit III**

**Attitudes and job Satisfaction:** What Are the Main Components of Attitudes, Does Behavior Always Follow from Attitudes, What Are the Major Job Attitudes, Job Satisfaction, Measuring Job Satisfaction, How Satisfied Are People in Their Jobs, What Causes Job Satisfaction, The Impact of Satisfied and Dissatisfied Employees on the Workplace

### **Unit IV**

**Emotions and Moods:** The Basic Emotions, The Basic Moods: Positive and Negative Affect, The Function of Emotions, Sources of Emotions and Moods, Emotional Labor, Affective Events Theory, Emotional Intelligence, The Case for EI, The Case Against EI, Emotion Regulation, OB Applications of Emotions and Moods.

### **Unit V**

**Perception and Decision Making:** Factors That Influence Perception, Making Judgments About Others, Attribution Theory, Common Shortcuts in Judging Others, Specific Applications of Shortcuts in Organizations, The Link Between Perception and Individual Decision Making, Decision Making in Organizations, The Rational Model, Bounded Rationality, and Intuition, Common Biases and Errors in Decision Making, Influences on Decision Making: Individual Differences and Organizational Constraints, Ethics in Decision Making, Three Ethical Decision Criteria, Improving Creativity in Decision Making.

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

**On successful completion of this course, the learner will be able to :**

CO1: To identify the various leadership styles and the role of leaders in a decision making process.

CO2: To explain organizational culture and describe its dimensions and to examine various organizational designs

CO3: Able to understand negative and positive emotions and moods.

CO4: Able to know the concept of Diversity in organization

CO5: Able to understand attitude and job satisfaction.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books**

1. “Organizational Behavior” Stephen P. Robbins —San Diego State University Timothy A. Judge —University of Notre Dame, 15<sup>th</sup> edition, Pearson.
2. McShane, S.L. and Von Glinow, M.A., Organizational Behaviour, New Delhi, Tata McGrawHill Publishing company ltd.

**Reference Books**

1. P. Jyothi, P. and Venkatesh, D.N., Human Resource Management, New Delhi, Oxford University Press.
2. Denhardt, R.B., Denhardt, J.V., and Aristigueta, M.P. (2009), Managing Human Behaviour in Public and Non-Profit Organizations, Second edition. California, Sage Publications.
3. Pynes, J.E. (2004). Human Resources Management for Public and Nonprofit Organizations, Second Edition. San Francisco, CA: Jossey- Bass Publishers.

**Course Name: Artificial Intelligence and Expert System**  
**Course Code: MCA173A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objective:**

1. To understand AI methodology and fundamentals; intelligent agents.
2. To understand search algorithms; game playing; supervised and unsupervised learning; decision tree learning.
3. To learn neural networks, nearest neighbor methods, dimensional reduction, clustering.
4. To understand kernel machines, support vector machines; uncertainty and probability theory.
5. To understand probabilistic reasoning in AI.

**Syllabus**

**Unit I**

**History & Knowledge Representation:** Artificial Intelligence: History and Applications, Intelligence, Communication, Learning, Artificial Intelligence, Progress of Artificial Intelligence, Modeling, Simulation, and AI, Intelligent Systems, Comparison of Various Intelligent Systems

Knowledge Representation: Reasoning, Issues, and Acquisition, Propositional Calculus, Predicate Calculus, Rule-based Knowledge Representation, Symbolic Reasoning Under Uncertainty, Basic Knowledge Representation Issues, Knowledge Acquisition

**Unit II**

**Heuristic Search & State Space Search:** Heuristic Search: Search as a Problem-solving Technique, Techniques for Heuristic Search, Admissibility, Monotonicity, Heuristic Classification, Intelligent Agents

State Space Search: Implementation and Applications, Strategies for State Space Search, Implementation of Graph Search, Search Based on Recursion, Production Systems, Natural

Language Processing , Applications of Search Techniques in Game Playing and Planning, Predicate Calculus and Planning

### **Unit III**

**Artificial Intelligence Problem-solving Languages:** PROLOG , Control Structures used in Rule-Based Systems , Search, Search Strategies Using Search Techniques , Implementation of Search, Construction and Debugging Methods for a Program, LISP, Problem-solving by LISP, Implementation of LISP , Comparison of LISP and PROLOG

### **Unit IV**

**Expert & Fuzzy Systems:** Expert Systems : Features ,Characteristics, Architecture, Goals, Basic Activities, Advantages, Stages in the Development of an Expert System, Probability-based Expert Systems, Expert System Tools , Applications

Fuzzy Systems : Foundation of Fuzzy Systems, Fuzzy Relations ,Arithmetic Operations of Fuzzy Numbers, Linguistic Descriptions and their Analytical Forms , De- fuzzification Methods, Fuzzy Logic in Control and Decision-making Applications,

### **Unit V**

**Swarm Intelligent Systems:** Swarm Intelligent Systems: Background of Ant Intelligent Systems, Importance of the Ant Colony Paradigm, Ant Colony Systems, Biological Ant Colony Systems, Artificial Ant Colony Systems, Development of the Ant Colony System, Applications of Ant Colony Intelligence

### **Course Outcomes:**

**On successful completion of this course students will be able to:**

CO1: Knowledge of what constitutes "Artificial" Intelligence and History of Artificial Intelligence.

CO2: Explain Implementations Applications related to Artificial Intelligence.

CO3: Implement Expert system technologies and Fuzzy logic theory.

CO4: Ability to apply Artificial Intelligence with neural network and genetic algorithm.

CO5: Explain the Swarm Artificial Intelligence techniques.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF  
PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books**

1. N. P. Padhy, "Artificial Intelligence and Intelligent Systems", 1st Edition, Oxford University Press , 2005.
2. Stuart Russell and Peter Norvig, "Artificial Intelligence A Modern Approach", 3rd Edition , Pearson Education, 2010.

**Reference Books**

3. Elaine Rich, Kevin Knight, Shivashankar B. Nair, "Artificial Intelligence", 3rd Edition, TataMcGraw Hill Edition, 2008.
4. SarojKaushik, "Artificial Intelligence", (1e), Cengage Learning Publications, 2011.
5. Don W. Patterson, "Introduction to Artificial Intelligence and Expert Systems", PHI Publication, 2006.



**Course Name: Computer Graphics**

**Course Code: MCA174A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objective:**

1. To provide comprehensive introduction about computer graphics system, design algorithms and two dimensional transformations.
2. To make the students familiar with techniques of clipping, three dimensional graphics and three dimensional transformations.
3. To be prepare students for activities involving in design, development and testing of modeling.
4. To make the students familiar with techniques of rendering, shading and animation.
5. To know about Modeling Transformations.

**Syllabus**

**Unit I**

**Computer Graphics Basics and Object Representation:** Historical Development of the Field, Major Issues and Concerns in Computer Graphics, Preliminaries: Basics of Graphics System, CRT Displays, Graphics Pipeline: Stages of Rendering Process, Role of Graphics Libraries. Categorization of Representation Techniques, Boundary Representation Techniques, Spline Representations, Space-partitioning Representation, Other Representations, Issues in Model Selection.

**Unit II**

**Modelling Transformations, Illumination, Lighting Models, and Intensity Representation:** Basic Transformations, Matrix Representation and Homogeneous Coordinate System, Composition of Transformations, Transformations in 3D ,Simple Lighting Model, Shading Models, Handling the Shadow Effect Intensity Representation.

### **Unit III**

**Color Models, Texture Synthesis, 3D Viewing and Clipping:** Physiology of Vision, Color Models, Texture Synthesis, 3D Viewing Transformation, Projection, Window-to-viewport Transformation, Clipping in 2D, 3D clipping.

### **Unit IV**

**Hidden Surface Removal and Rendering:** Types of Methods, Application of Coherence, Back Face Elimination, Depth (Z) Buffer Algorithm, A-Buffer Algorithm, Depth Sorting (Painter's) Algorithm, Warnock's Algorithm, Octree Methods, Scan Conversion of a Line Segment, DDA Algorithm, Bresenham's Algorithm, Circle Scan Conversion, Midpoint Algorithm, Fill Area Scan Conversion, Seed Fill Algorithm, Flood Fill Algorithm, Scan Line Polygon Fill Algorithm, Character Rendering, Anti-aliasing, Aliasing and Signal Processing, Pre-filtering or Area Sampling, Gupta-Sproull Algorithm, Super Sampling.

### **Unit V**

**Computer Animation , Multimedia and Hypermedia:** Generic Architecture, Input and Output of Graphics System, GPU and Shader Programming, Graphics Software and OpenGL, Traditional Animation Tech, Keyframing, Motion Capture, Physically based Methods and Procedural Techniques. Hypermedia, Multimedia Authoring, Components of Multimedia, Data Compression Standards

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

**On successful completion of this course students will be able to:**

CO1: Students will get the concepts of Graphics display devices, different types of graphics drawing algorithms.

CO2: Students will get the concepts of 2D and 3D Geometrical Transformations

CO3: Students will get the concepts of Color models and clipping.

CO4: Students will get the concepts of Hidden Line/surface elimination techniques

CO5: Student will get the concepts of creating animations and multimedia.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Computer Graphics, Oxford University, Bhattacharya, December 2015.
2. “Mathematical Elements of Computer Graphics” by Kogin Adams

**Reference Books**

1. “Computer Graphics” by John F Hughes
2. “Computer Graphics” by Desai and Apurva A
3. “Computer Graphics: A Programming Approach” by Steven Harrington

**Course Name: Management Information System**

**Course Code: MCA175A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objectives**

1. To get the knowledge about the important management concepts & their application, to have an insight of various functional departments in an organization.
2. To discuss the importance of security, privacy, and ethical issues as they relate to information systems.
3. To identify some of the strategies employed to lower costs and improve service.
4. To know about the designing and development of MIS
5. To know about Role of Decision Support System in MIS, Decision Support Models,

**Syllabus**

**Unit I**

**Introduction to Information System:** Why information system, Classification of Information, Key aspects of Management, Definitions, roles and features of MIS, Structure of Management Information System, Benefits of Management Information System, Limitations of Management Information System, Management Support Systems

**Unit II**

**MIS Development:** Overview of design of an information system, The role and tasks of systems analysts, Tools used by system analyst in designing Information system, MIS Requirement Analysis, MIS Requirement Specification, Feasibility Analysis & Report, MIS Development Models, MIS Design & Development Phase

**Unit III**

**Decision Support System in MIS:** Managerial Decision Making, Types of Decisions, Definition and characteristics of DSS, Types of Decision Support Systems, Components of DSS, Role of Decision Support System in MIS, Decision Support Models, Risks of DSS in MIS,

#### **Unit IV**

**Data warehousing and mining:** Technology of Information Systems, Unified Communication and Networks, Database and Client Server Architecture. Data Warehouse: Architecture to Implementation E-Business Technology, Data Mining Concepts and Applications, Tools and Techniques, Text Mining, Web Mining

#### **Unit V**

**MIS Security:** MIS Security Risks, Threats & Vulnerability, Assessing Risks. Common MIS Controls (Physical, Electronic, Software, Management Controls), MIS Threats (Natural Disasters, Employee Errors, Computer Crime, Fraud, Abuse, Program Bugs), Information Security and control concepts-Access controls, QA and QC concepts with respect to the processes of various functional areas of management, social and security issues related to MIS, Control, Audit and Security of MIS.

#### **Course Outcomes (Cos):**

**Upon successful completion of this subject students should be able to:**

CO1: Understand the roles and features of MIS, Structure of Management Information System, and Benefits of Management Information System.

CO2: Know about the designing and development of MIS, MIS Requirement Specification, and Feasibility Analysis & Report.

CO3: To identify MIS Security Risks, Threats & Vulnerability and Assessing Risks.

CO4: To identify Database and Client Server Architecture. Data Warehouse: Architecture to implementation.

CO5: Know about Role of Decision Support System in MIS, Decision Support Models,

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

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

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Management Information Systems: Waman S Jawadekar, Tata McGraw-Hill
2. Management Information Systems S. Sadagopan, PHI learning PVT Ltd.

**Reference Books -**

1. Management Information Systems, Davis, Tata McGraw-Hill
2. Decision Support and Expert Systems: Management Support Systems, Efraim Turban, Prentice Hall
3. Dr. C.B Gupta “Management concepts & practices” S.Chand& Sons, 2009.

**Course Name: Robotics Process Automation**

**Course Code: MCA186A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Syllabus**

**Unit-1**

Programming Basics & Recap:- Programming Concepts Basics, Understanding the application, Basic Web Concepts, Protocols, Email Clients, Data Structures, Data Tables, Algorithms, Software.

Processes, Software Design, SDLC:- Programming Concepts Basics – 2 Scripting, .Net Framework, .Net, Fundamentals, XML, Control structures and functions, XML, HTML, CSS, Variables & Arguments.

RPA Basics:- History of Automation, What is RPA, RPA vs Automation, Processes & Flowcharts, Programming Constructs in RPA, What Processes can be Automated, Types of Bots, Workloads which can be automated.

**Unit- II**

RPA Advanced Concepts :Standardization of processes, RPA Development methodologies, Difference from SDLC, Robotic control flow architecture, RPA business case, RPA Team, Process Design Document/Solution Design Document, Industries best suited for RPA, Risks & Challenges with RPA, RPA and emerging ecosystem

Installation :Installing Studio community edition, The User Interface, Keyboard Shortcuts, About Updating, About Automation Projects, Introduction to Automation Debugging, Managing Activation Packages, Reusing Automations Library, Installing the Chrome Extension,

Installing the Firefox Extension, Connecting your project to a source control system, Activities Guide

Variables :Managing Variables, Naming Best Practices, The Variables Panel, Generic Value Variables, Text Variables, True or False Variables, Number Variables, Array Variables, Date and Time Variables, Data Table Variables, Managing Arguments, Naming Best Practices, The Arguments Panel, Using Arguments, About Imported Namespaces, Importing New Namespaces

**Unit-III**

Control Flow :Control Flow Introduction,If Else Statements,Loops,Advanced Control Flow,Sequences,Flowcharts>About Control Flow,Control Flow Activities,The Assign Activity,The Delay Activity,The Do While Activity,The If Activity,The Switch Activity,The While Activity,The For Each Activity,The Break Activity

Data Manipulation :Data Manipulation Introduction,Scalar variables, collections and Tables,TextManipulation,DataManipulation,Gathering and Assembling Data.

Recording and Advanced UI Interaction :Recording Introduction,Basic and Desktop Recording, WebRecording, Input/Output Methods, ScreenScraping,DataScraping,Scraping advanced techniques. Selectors: Selectors, Defining and Assessing Selectors,Customization, Debugging, Dynamic Selectors, Partial Selectors, RPA Challenge.

#### **Unit-IV**

Advanced Automation concepts and techniques: Image, Text & Advanced Citrix Automation: Introduction to Image & Text ,Automation,Image based automation,Keyboard based automation,InformationRetrieval,Advanced Citrix Automation challenges,BestPractices,Using tab for Images,Starting Apps.

Excel Data Tables & PDF :Data Tables in RPA,Excel and Data Table basics,Data Manipulation in excel,Extracting Data from PDF,Extracting a single piece of data,Anchors,Using anchors in PDF

Email Automation :Email Automation,Incoming Email automation,Sending Email automation  
Exceptional Handling & Best Practices

Debugging and Exception Handling :Debugging Tools,Strategies for solving issues,Catching errors

Project Organization:What is project organization,Bestpractices,Avoidingpitfalls,Invoke Activity

#### **Unit-V**

Introduction to Orchestrator: Orchestrator: Tenants, Authentication, Users,Roles, Robots, Environments, Queues &Transactions, Schedules.

Emerging and Future Trends in IT: Emerging and Future Trends in IT: Artificial Intelligence, MachineLearning, Agentawareness,Natural Language Processing, Computer Vision

Capstone Project: Real life case studies which can be used to apply the concepts learnt during the course. The projects shall test student's skills right from process transformation and documentation to the design and development of the actual robot.

**\*\* All the practical topics are covered in respective Lab sessions**



## **List of Courses for Specialization in Cyber Security**

### **Certified Network Defender CND**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	0	2	5

#### **Unit – I**

##### 1. Computer Network and Defense Fundamentals

- Understanding computer network
- Describing OSI and TCP/IP network Models
- Comparing OSI and TCP/IP network Models
- Understanding different types of networks
- Describing various network topologies
- Understanding various network components
- Explaining various protocols in TCP/IP protocol stack
- Explaining IP addressing concept
- Understanding Computer Network Defense (CND)
- Describing fundamental CND attributes
- Describing CND elements
- Describing CND process and approaches

##### Network Security Threats, Vulnerabilities, and Attacks

- Understanding threat, attack, and vulnerability
- Discussing network security concerns
- Reasons behind network security concerns
- Effect of network security breach on business continuity
- Understanding different types of network threats
- Understanding different types of network security vulnerabilities
- Understanding different types of network attacks
- Describing various network attacks

## Network Security Controls, Protocols, and Devices

- Understanding fundamental elements of network security
- Explaining network access control mechanism
- Understanding different types of access controls
- Explaining network Authentication, Authorization and Auditing (AAA) mechanism
- Explaining network data encryption mechanism
- Describing Public Key Infrastructure (PKI)
- Describing various network security protocols
- Describing various network security devices

## **Unit – II**

### Network Security Policy Design and Implementation

- Understanding security policy
- Need of security policies
- Describing the hierarchy of security policy
- Describing the characteristics of a good security policy
- Describing typical content of security policy
- Understanding policy statement
- Describing steps for creating and implementing security policy
- Designing of security policy
- Implementation of security policy
- Describing various types of security policy
- Designing of various security policies
- Discussing various information security related standards, laws and acts

### Physical Security • Understanding physical security

- Importance of physical security
- Factors affecting physical security
- Describing various physical security controls

- Understanding the selection of Fire Fighting Systems
- Describing various access control authentication techniques
- Understanding workplace security
- Understanding personnel security
- Describing Environmental Controls
- Importance of physical security awareness and training

#### Host Security • Understanding host security

- Understanding the importance of securing individual hosts
- Understanding threats specific to hosts
- Identifying paths to host threats
- Purpose of host before assessment
- Describing host security baselining
- Describing OS security baselining
- Understanding and describing security requirements for different types of servers
- Understanding security requirements for hardening of routers
- Understanding security requirements for hardening of switches
- Understanding data security concerns when data is at rest, in use, and in motion
- Understanding virtualization security

### **Unit – III**

#### Secure Firewall Configuration and Management

- Understanding firewalls
- Understanding firewall security concerns
- Describing various firewall technologies
- Describing firewall topologies
- Appropriate selection of firewall topologies
- Designing and configuring firewall ruleset
- Implementation of firewall policies
- Explaining the deployment and implementation of firewall
- Factors to consider before purchasing any firewall solution

- Describing the configuring, testing and deploying of firewalls
- Describing the management, maintenance and administration of firewall implementation
- Understanding firewall logging
- Measures for avoiding firewall evasion
- Understanding firewall security best practices

#### Secure IDS Configuration and Management

- Understanding different types of intrusions and their indications
- Understanding IDPS
- Importance of implementing IDPS
- Describing role of IDPS in network defense
- Describing functions, components, and working of IDPS
- Explaining various types of IDS implementation
- Describing staged deployment of NIDS and HIDS
- Describing fine-tuning of IDS by minimizing false positive and false negative rate
- Discussing characteristics of good IDS implementation
- Discussing common IDS implementation mistakes and their remedies
- Explaining various types of IPS implementation
- Discussing requirements for selecting appropriate IDPS product
- Technologies complementing IDS functionality

#### Secure VPN Configuration and Management

- Understanding Virtual Private Network (VPN) and its working
- Importance of establishing VPN
- Describing various VPN components
- Describing implementation of VPN concentrators and its functions
- Explaining different types of VPN technologies
- Discussing components for selecting appropriate VPN technology
- Explaining core functions of VPN
- Explaining various topologies for implementation of VPN

- Discussing various VPN security concerns
- Discussing various security implications to ensure VPN security and performance

## **Unit – IV**

### **Wireless Network Defense**

- Understanding wireless network
- Discussing various wireless standards
- Describing various wireless network topologies
- Describing possible use of wireless networks
- Explaining various wireless network components
- Explaining wireless encryption (WEP, WPA, WPA2) technologies
- Describing various authentication methods for wireless networks
- Discussing various types of threats on wireless networks
- Creation of inventory for wireless network components
- Appropriate placement of wireless Access Point (AP)
- Appropriate placement of wireless antenna
- Monitoring of wireless network traffic
- Detection and locating of rogue access points
- Prevention of wireless network from RF interference
- Describing various security implications for wireless network

### **Network Traffic Monitoring and Analysis**

- Understanding network traffic monitoring
- Importance of network traffic monitoring
- Discussing techniques used for network monitoring and analysis
- Appropriate position for network monitoring
- Connection of network monitoring system with managed switch
- Understanding network traffic signatures
- Baseline for normal traffic
- Discussing the various categories of suspicious traffic signatures
- Various techniques for attack signature analysis

- Understanding Wireshark components, working and features
- Demonstrating the use of various Wireshark filters
- Demonstrating the monitoring LAN traffic against policy violation
- Demonstrating the security monitoring of network traffic
- Demonstrating the detection of various attacks using Wireshark
- Discussing network bandwidth monitoring and performance improvement

#### Network Risk and Vulnerability Management

- Understanding risk and risk management
- Key roles and responsibilities in risk management
- Understanding Key Risk Indicators (KRI) in risk management
- Explaining phase involves in risk management
- Understanding enterprise network risk management
- Describing various risk management frameworks
- Discussing best practices for effective implementation of risk management
- Understanding vulnerability management
- Explaining various phases involved in vulnerability management
- Understanding vulnerability assessment and its importance
- Discussing requirements for effective network vulnerability assessment
- Discussing internal and external vulnerability assessment
- Discussing steps for effective external vulnerability assessment
- Describing various phases involve in vulnerability assessment
- Selection of appropriate vulnerability assessment tool
- Discussing best practices and precautions for deploying vulnerability assessment tool
- Describing vulnerability reporting, mitigation, remediation and verification

### **Unit – V**

#### Data Backup and Recovery

- Understanding data backup
- Describing the data backup plan
- Describing the identification of data to backup

- Determining the appropriate backup medium for data backup
- Understanding RAID backup technology and its advantages
- Describing RAID architecture
- Describing various RAID levels and their use
- Selection of appropriate RAID level
- Understanding Storage Area Network (SAN) backup technology and its advantages
- Best practices of using SAN
- Understanding Network Attached Storage (NAS) backup technology and its advantages
- Describing various types of NAS implementation

#### Network Incident Response and Management

- Understanding Incident Handling and Response (IH&R)
- Roles and responsibilities of Incident Response Team (IRT)
- Describing role of first responder
- Describing first response activities for network administrators
- Describing Incident Handling and Response (IH&R) process
- Understanding forensic investigation
- People involved in forensics investigation
- Describing forensics investigation methodology

## **Certified Ethical Hacker CEH**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	0	0	3

### **Unit – I**

#### Background Network and Communication Technologies

- Networking technologies (e.g., hardware, infrastructure)
- Web technologies (e.g., web 2.0, skype)
- Systems technologies
- Communication protocols
- Telecommunication technologies
- Mobile technologies (e.g., smartphones)
- Wireless terminologies
- Cloud computing
- Cloud deployment models

#### Information Security Threats and Attack Vectors

- Malware (e.g., Trojan, virus, backdoor, worms)
- Malware operations
- Information security threats and attack vectors
- Attacks on a system (e.g., DoS, DDoS, session hijacking, webserver and web application attacks, SQL injection, wireless threats)
- Botnet
- Cloud computing threats and attacks
- Mobile platform attack vectors
- Cryptography attacks

#### Information Security Technologies

- Information security elements
- Information security management (e.g. IA, Defense-in-Depth, incident management)



- Security trends
- Hacking and ethical hacking
- Vulnerability assessment and penetration testing
- Cryptography
- Encryption algorithms
- Wireless encryption
- Bring Your Own Device (BYOD)
- Backups and archiving (e.g., local, network)
- IDS, firewalls, and honeypots

## **Unit – II**

### Analysis / Assessment Information Security Assessment and Analysis

- Data analysis
- Systems analysis
- Risk assessments
- Vulnerability assessment and penetration testing
- Technical assessment methods
- Network sniffing
- Malware analysis

### Information Security Assessment Process

- Footprinting
- Scanning (e.g., Port scanning, banner grabbing, vulnerability scanning, network discovery, proxy chaining, IP spoofing)
- Enumeration
- System hacking (e.g., password cracking, privilege escalation, executing applications, hiding files, covering tracks)

## **Unit – III**

### Security Information Security Controls

- Systems security controls

- Application/file server
- IDS
- Firewalls
- Cryptography
- Disk Encryption
- Network security
- Physical security
- Threat modeling
- Biometrics
- Wireless access technology (e.g., networking, RFID, Bluetooth)
- Trusted networks
- Privacy/confidentiality (with regard to engagement)

#### Information Security Attack Detection

- Security policy implications
- Vulnerability detection
- IP Spoofing detection
- Verification procedures (e.g., false positive/negative validation)
- Social engineering (human factors manipulation)
- Vulnerability scanning
- Malware detection
- Sniffer detection
- DoS and DDoS detection
- Detect and block rogue AP
- Evading IDS (e.g., evasion, fragmentation)
- Evading Firewall (e.g., firewalking, tunneling)
- Honeypot detection
- Steganalysis

#### Information Security Attack Prevention

- Defend against web server attacks

- Patch management
- Encoding schemes for web application
- Defend against web application attacks
- Defend against SQL injection attacks
- Defend against wireless and Bluetooth attacks
- Mobile platforms security
- Mobile Device Management (MDM)
- BYOD Security
- Cloud computing security

### **Unit – IV**

#### **Tools / Systems / Programs Information Security Systems**

- Network/host based intrusion
- Boundary protection appliances
- Access control mechanisms (e.g., smart cards)
- Cryptography techniques (e.g., IPsec, SSL, PGP)
- Domain name system (DNS)
- Network topologies
- Subnetting
- Routers / modems / switches
- Security models
- Database structures

#### **Information Security Programs**

- Operating environments (e.g., Linux, Windows, Mac)
- Anti-malware systems and programs (e.g., anti-keylogger, anti-spyware, anti-rootkit, anti-trojan, anti-virus)
- Wireless IPS deployment
- Programming languages (e.g. C++, Java, C#, C)
- Scripting languages (e.g., PHP, Javascript)

## Information Security Tools

- Network/wireless sniffers (e.g., Wireshark, Aircrack-ng)
- Port scanning tools (e.g., Nmap, Hping)
- Vulnerability scanner (e.g., Nessus, Qualys, Retina)
- Vulnerability management and protection systems (e.g., Foundstone, Ecora)
- Log analysis tools
- Exploitation tools
- Footprinting tools (e.g., Maltego, FOCA, Recon-ng)
- Network discovery tools (e.g., Network Topology Mapper)
- Enumeration tools (e.g., SuperScan, Hyena, NetScanTools Pro)
- Steganography detection tools
- Malware detection tools
- DoS/DDoS protection tools
- Patch management tool (e.g., MBSA)
- Webserver security tools
- Web application security tools (e.g., Acunetix WVS)
- Web application firewall (e.g., dotDefender)
- SQL injection detection tools (e.g., IBM Security AppScan)
- Wireless and Bluetooth security tools
- Android, iOS, Windows Phone OS, and BlackBerry device security tools
- MDM Solutions
- Mobile Protection Tools
- Intrusion Detection Tools (e.g., Snort)
- Hardware and software firewalls (e.g., Comodo Firewall)
- Honeypot tools (e.g., KFSensor)
- IDS/Firewall evasion tools (e.g., Traffic IQ Professional)
- Packet fragment generators
- Honeypot Detection Tools
- Cloud security tools (e.g., Core CloudInspect)
- Cryptography tools (e.g., Advanced Encryption Package)

- Cryptography toolkit (e.g., OpenSSL)
- Disk encryption tools
- Cryptanalysis tool (e.g., CrypTool)

## **Unit – V**

### Procedures / Methodology Information Security Procedures

- Cryptography
- Public key infrastructure (PKI)
- Digital signature and Pretty Good Privacy (PGP)
- Security Architecture (SA)
- Service oriented architecture
- Information security incident
- N-tier application design
- TCP/IP networking (e.g., network routing)
- Security testing methodology

### Information Security Assessment Methodologies

- Web server attack methodology
- Web application hacking methodology
- SQL injection methodology and evasion techniques
- SQL injection evasion techniques
- Wireless and Bluetooth hacking methodology
- Mobile platform (Android, iOS, Windows

### Phone OS, and BlackBerry) hacking methodology

- Mobile Rooting and Jailbreaking

### Regulation / Policy Information Security Policies/ Laws/Acts

- Security policies
- Compliance regulations (e.g., PCI-DSS,SOX)

### Ethics of Information Security

- Professional code of conduct
- Appropriateness of hacking

**Certified Application Security Engineer CASE**

**JAVA**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
4	0	2	4

**Course Outline: Certified Application Security Engineer (CASE) Java**

**Unit – I**

Understanding Application Security,  
Threats and Attacks,  
Security Requirements Gathering

**Unit – II**

Secure Application Design and Architecture,  
Secure Coding Practices for Input Validation,

**Unit – III**

Secure Coding Practices for Authentication and Authorization,  
Secure Coding Practices for Cryptography

**Unit – IV**

Secure Coding Practices for Session Management,  
Secure Coding Practices for Error Handling

**Unit – V**

Static and Dynamic Application Security Testing (SAST & DAST),  
Secure Deployment and Maintenance

**Certified Application Security Engineer CASE**

**JAVA**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
4	0	2	4

**Course Outline: Certified Application Security Engineer (CASE) .Net**

**Unit – I**

Understanding Application Security,  
Threats, and Attacks,  
Security Requirements Gathering

**Unit – II**

Secure Application Design and Architecture,  
Secure Coding Practices for Input Validation

**Unit – III**

Secure Coding Practices for Authentication and Authorization,  
Secure Coding Practices for Cryptography

**Unit – IV**

Secure Coding Practices for Session Management,  
Secure Coding Practices for Error Handling

**Unit – V**

Static and Dynamic Application Security Testing (SAST & DAST),  
Secure Deployment and Maintenance

\*\*\*\*\*