



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

**Faculty of IT & Computer Application**

**Scheme & Syllabi**

**of**

**Master of Computer Applications (MCA)**

**(Specialization in Artificial Intelligence and Machine Learning)**

**( In Association with Samatrix)**

**Academic Programme**

**July2020-22**

## MCA

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

- 1. Minimum Credits required = 102 Credits**
- 2. Total Relaxation = 10 Credits**
- 3. No relaxation in subjects of Core and Foundation.**
- 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.**

| <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<br>(Hr.) | T<br>(Hr.) | P<br>(Hr.) | Credits   | Type |
|--|---|------------|------------|------------|-----------|------|
| MCA119A  | Programming in C++                                | 3          | 0          | 0          | 3         | CORE |
| MCA120B  | Computer Networks                                 | 4          | 0          | 0          | 4         | CORE |
| MCA121B  | Advance Data Structures and Algorithms            | 3          | 0          | 0          | 3         | CORE |
| <b>MCAAI101<br/>(AI &amp; ML<br/>Specialization)</b> | Data Analysis using Python                        | 2          | 0          | 2          | 4         | CORE |
| <b>MCAAI102<br/>(AI &amp; ML<br/>Specialization)</b> | Probabilistic Modelling and Reasoning with Python | 2          | 0          | 2          | 4         | CORE |
| MCA123B  | Business Communication Skills                     | 2          | 0          | 0          | 2         | F    |
| 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    |
| MCA176A  | Designing Lab ( Photoshop and CorelDraw)          | 0          | 0          | 2          | 2         | S    |
|  | <b>Total</b>                                      | <b>16</b>  | <b>0</b>   | <b>12</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<br>(Hr.) | T<br>(Hr.) | P<br>(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<br>Management Systems   | 3          | 0          | 0          | 3         | CORE |
| <b>MCAAI103</b><br><b>(AI &amp; ML</b><br><b>Specialization</b> | Machine Learning and Pattern<br>Recognition  | 2          | 0          | 2          | 4         | CORE |
| <b>MCAAI104</b><br><b>(AI &amp; ML</b><br><b>Specialization</b> | Machine Learning Practical<br>with Python, Scikit-learn,<br>Matplotlib, TensorFlow | 2          | 0          | 2          | 4         | CORE |
| MCA133A   | Advance Java Lab   | 0          | 0          | 2          | 2         | S    |
| MCA134B   | PHP & MYSQL Lab  | 0          | 0          | 2          | 2         | S    |
| MCA124B   | Advance Database<br>Management Systems Lab   | 0          | 0          | 2          | 2         | S    |
| MCA136A   | Minor Project  | 0          | 0          | 2          | 2         | S    |
|   | <b>Total</b>   | <b>16</b>  | <b>0</b>   | <b>12</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<br>(Hr.) | T<br>(Hr.) | P<br>(Hr.) | Credits   | Type |
|---|--|------------|------------|------------|-----------|------|
| MCA129A   | Accounting Principles and Practices                | 3          | 0          | 0          | 3         | ID   |
| MCA183A   | Agile Software Development                         | 3          | 0          | 0          | 3         | F    |
| MCA138A   | Information Security & Cyber Law                   | 3          | 0          | 0          | 3         | CORE |
| MCA184A   | Data Science & Analytics                           | 3          | 0          | 0          | 3         | S    |
| MCA185A   | Data Science & Analytics Lab                       | 0          | 0          | 2          | 2         | S    |
| MCA177A   | Cloud Computing Lab                                | 0          | 0          | 2          | 2         | S    |
| <b>MCAAI105</b><br><b>(AI &amp; ML</b><br><b>Specialization</b> | Neural Networks and Deep Learning (Vision and NLP) | 2          | 0          | 2          | 4         | CORE |
| <b>MCAAI106</b><br><b>(AI &amp; ML</b><br><b>Specialization</b> | Data Visualization                                 | 2          | 0          | 2          | 4         | CORE |
| <b>MCAAI107</b><br><b>(AI &amp; ML</b><br><b>Specialization</b> | Project  | 0          | 0          | 4          | 4         | CORE |
|   | <b>Total</b>                                       | <b>16</b>  | <b>0</b>   | <b>12</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 |

# Semester I

**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, Files 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**

| L (Hr.) | T/P (Hr.) | Pr (Hr.) | Credits |
|---------|-----------|----------|---------|
| 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,3 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 NarasimhaKarumanchi
3. “Data Structures and Program Design In C” by Robert L Kruse and Bruce P Leung

**Course Name: Statistical Computing**

**Course Code: MCA122A**

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

## Course Objectives

1. To help in design, data collection plans, analyze data appropriately and interpret and draw conclusions from analyses.
2. To be able to exploit central objective of the undergraduate major in Statistics is to equip students with consequently requisite quantitative skills that they can employ and build on in flexible ways.
3. To learn the concepts and tools for working with data and have experience in analyzing real data.
4. To understand the fundamentals of probability theory, statistical reasoning, inferential methods, statistical modeling and its limitations.
5. To exploratory analysis of data by graphical and other means.

## Syllabus

### Unit I

**Basic Statistics:** Definition, Function & Scope of Statistics. Measures of Central Tendency: Arithmetic Mean, Weighted A.M., Median, Mode, Geometric & Harmonic Mean and Their Merits & Demerits. Measures of dispersion. Frequency distributions.

## **Unit II**

**Measures of Variation:** Range, The Interquartile Range or Quartile Deviation, Average (Mean), Deviation Standard Deviation, Coefficient of Variation, Skewness, Moments & Kurtosis.

Sampling: Theory of sampling, population and sample Survey methods and estimation  
Statistical inference, testing of hypothesis.

## **Unit III**

**Correlation Analysis:** Introduction, Karl Pearson's Coefficient of Correlation, Rank Correlation Coefficient.

## **Unit IV**

**Regression Analysis:** Difference between Correlation & Regression, Regression Lines, Regression Equations, Regressions Coefficient.

## **Unit V**

**Sampling Distribution:** Chi Square ( $\chi^2$ ) Distribution and Its Properties, Chi - Square Test, Application of Chi -Square Distribution: Chi-Square Test for Population Variance, ChiSquare Test of Goodness of Fit, Independence of Attributes, T- Distribution & Its Properties, Application of T - Distribution to Testing Hypothesis About Population Mean, Difference Between Two Means, Correlation Coefficient, F- Distribution.

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

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

CO1: Use the computer to conduct a statistical analysis of data, including how to acquire, clean and organize data, analyze data using computationally intensive statistical methods.

CO2: To acquire skills of regression equation and coefficient.

CO3: To test hypothesis and methodology such as sampling, goodness-of-fit testing, analysis of variance and least squares estimation.

CO4: To design data collection plans, analyze data appropriately and apply Chi-Square Testing.

CO5: To simplify Correlation Coefficient with different 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   |     |     |     |     |     |     | M   |      |      |      |
| CO2            |                 |     | L   |     | H   |     |     |     |     |      | M    |      |
| CO3            |                 |     |     |     |     | M   |     | H   |     |      |      |      |
| CO4            |                 | M   |     | H   |     |     |     |     |     |      |      | M    |
| CO5            | H               |     |     |     |     |     | L   |     |     | M    |      |      |

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

**Text Books**

1. Affi, A.A., "Statistical Analysis: A Computer Oriented Approach". Academic Press, New York, 1979.
2. Hogg, R. v..Et. Al., "Introduction to Mathematical Statistics", American Publishing, New York. 1980.

**Reference Books**

1. Bailer, A. J. (2010). Statistical Programming in SAS Cary, NC: SAS Institute Inc. Delwiche, L. and Slaughter, S. (2008).
2. The Little SAS Book: A Primer, Fourth Edition. Cary, NC: SAS Institute Inc. SAS Institute Inc. (2011). SAS Inc. 9.3



3. Language Reference: Concepts. Cary, NC: SAS Institute

**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 C++ Lab**

**Course Code: MCA125A**

| L (Hr.) | T/P (Hr.) | Pr (Hr.) | Credits |
|---------|-----------|----------|---------|
| 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:  $1/1+2^2/2+3^3/3+\dots$
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

**Course Name: Designing Lab (Photoshop and CorelDraw)**

**Course Code: MCA176A**

| <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. This exercise can be solved using the following functions: Selection Tools, Copy, Cut, Paste, Move Tool.
2. This exercise can be solved using the following functions: Lasso- and Polygonal Lasso Selection Tools, Copy, Paste Into, Move Tool, Zoom Tool, Transform.
3. This exercise can be solved using the following functions: Quick Select Tool (or Magic Wand Tool), Invert Selection, Copy, Paste.
4. This exercise can be solved using the following functions: Paint Bucket Tool, Color Picker, Brush Tool.
5. This exercise can be solved using the following functions: New Picture, Brush Tool, Layers, Eraser Tool.
6. Creating the master page in CorelDraw
7. Creating pie charts in CorelDraw
8. Creating flow charts in CorelDraw
9. Creating interactive navigational buttons in CorelDraw
10. Publishing your presentation as a full screen PDF in CorelDraw

## 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:**

| Course Outcomes | Program Outcomes |     |     |     |     |     |     |     |     |      |      |      |
|-----------------|------------------|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|
|                 | 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**

| L (Hr.) | T/P (Hr.) | Pr (Hr.) | Credits |
|---------|-----------|----------|---------|
| 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

| L (Hr.) | T/P (Hr.) | Pr (Hr.) | Credits |
|---------|-----------|----------|---------|
| 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: Advance Database Management System Lab**

**Course Code: MCA124B**

| <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. Create a relational database that contains the some tables and insert records into these tables. Solve the different SQL Queries.
2. Create the 3 structured record data type named as address\_type(street, city, state, pin\_code), person\_type(firstname, lastname, DOB) and business\_type( title, company). Create 2 tables based on person\_type and business\_type. Create one more table which is referred by first two tables. ( Reference type)
3. Write a program to enter a number and find the factorial of the number.
4. Write a code to create a type with an array of number 10.
5. Write a PL/SQL code to take the input from the user for 3 subjects and calculate total marks and percentage.
6. Write a program to create a function for add two numbers and call it by passing the values as an argument.
7. Write a program to enter two number and find greater number using function
8. Write a program to create a PL/SQL function for count the number of records in any existing table.
9. Write a PL/SQL code to create a table using the already created array type and insert 5 records.
10. Write a PL/SQL code to enter two numbers and perform the arithmetic operations. (Addition, Subtraction, Multiplication, Division)
11. Write a PL/SQL code to retrieve the employee name and city from employee database of an employee whose number is input by the user. (Create a Employee table with the field named as emp\_no, employee name, street, city)
12. Write a program to create a table with at least 3 fields and create a procedure for insert data in the existing table. Insert 5 records through this procedure.
13. Write a PL/SQL block to calculate the incentive of an employee whose ID is 110.
14. Write a PL/SQL block to adjust the salary of the employee whose ID 122.
15. Write a PL/SQL block to describe the usage of LIKE operator including wildcard characters and escape character.
16. Write a PL/SQL procedure to calculate the incentive on a target achieved and display the message either the record updated or not.
17. Write a PL/SQL program to check whether a number is even or odd.



18. Write a PL/SQL procedure to calculate the incentive on a specific target otherwise a general incentive to be paid using IF-THEN-ELSE.
19. Write a PL/SQL program to check whether a date falls on weekend i.e. SATURDAY or SUNDAY.
20. Write a PL/SQL program to display the description against a grade using CASE statement.
21. Write a PL/SQL program to check whether a given number is positive, negative or zero.
22. Write a PL/SQL program to check whether a given character is letter or digit.
23. Write a PL/SQL program to convert a temperature in scale Fahrenheit to Celsius and vice versa.
24. Write a program in PL/SQL to print the value of a variable inside and outside a loop using LOOP EXIT statement.
25. Write a program in PL/SQL to show the uses of nested loop.
26. Write a program in PL/SQL using loop with CONTINUE statement.
27. Write a program in PL/SQL to print 1st n numbers.
28. Write a program in PL/SQL to check whether a number is prime or not using goto statement with for loop.
29. Write a program in PL/SQL to insert records from one table to another.
30. Write a program in PL/SQL to show the uses of CURVAL and NEXTVAL with a sequence name.
31. Write a program in PL/SQL to find the number of rows effected by the use of SQL%ROWCOUNT attributes of an implicit cursor
32. Write a program in PL/SQL to show the uses of implicit cursor without using any attribute.
33. Write a program in PL/SQL to show the uses of SQL%FOUND to determine if a DELETE statement affected any rows.
34. Write a program in PL/SQL to show the uses of SQL%NOTFOUND to determine if a UPDATE statement affected any rows.
35. Write a program in PL/SQL to retrieve the records from the employees table and display them using cursors.
36. Write a program in PL/SQL to declare a record datatype with same datatype of tables using %TYPE attribute.
37. Write a program in PL/SQL to create an implicit cursor and explicit cursor with for loop.

38. Write a program in PL/SQL to FETCH multiple records and more than one columns from the same table.
39. Write a program in PL/SQL to FETCH records with nested Cursors using Cursor FOR Loops.
40. Write a function incr\_salary, which takes two parameters, employee ssn and the percentage by which the salary needs to be raised and returns the new salary.
41. Write a procedure Get\_On\_Hand that has one IN and one OUT parameter. This procedure should take in equipment number and send the quantity on hand for this equipment through the out parameter.
42. Write a procedure called emp\_dept which has one IN and one OUT parameter. This procedure should take in employee ssn and send the name of the department he is working for by getting it from the department table.
43. Write a trigger, which checks the quantity on hand in the equipment table for every update and displays an error message if the quantity on hand falls below 2.
44. Write a trigger, which fires before the update of employee table for each row and checks if the new salary is below 20000, if it is then it raises an error.

**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 Outcome | Program 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 Accounting (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 Securityl, Thomson Learning, Boston, 2007.
2. Bruce Schneier, —Applied Cryptographyl, 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

# List of Courses for Specialization in Artificial Intelligence and Machine Learning

## DATA ANALYSIS USING PYTHON

|                          |
|--------------------------|
| Credit: 2-0-1            |
| No of Hours : 4 Per Week |

*Objectives: The objective of this course is to teach students the concepts of Python Programming Language with Libraries*

### UNIT – I

**Python programming Basic:** Python interpreter, IPython Basics, Tab completion, Introspection, %run command, magic commands, matplotlib integration, python programming, language semantics, scalar types. Control flow

### UNIT – II

**Data Structure, functions, files:** tuple, list, built-in sequence function, dict, set, functions, namespace, scope, local function, returning multiple values, functions are objects, lambda functions, error and exception handling, file and operation systems

### UNIT – III

**NumPy: Array and vectorized computation:** Multidimensional array object. Creating ndarrays, arithmetic with numpy array, basic indexing and slicing, Boolean indexing, transposing array and swapping axes, universal functions, array-oriented programming with arrays, conditional logic as arrays operations, file input and output with array

### UNIT – IV

**Pandas:** Pandas data structure, series, DataFrame, Index Object, Reindexing, dropping entities from an axis, indexing, selection and filtering, integer indexes, arithmetic and data alignment, function application and mapping, sorting and ranking, correlation and covariance, unique values, values controls and membership, reading and writing data in text format



## **UNIT -V**

**Visualization with Matplotlib:** Figures and subplots, colors, markers, line style, ticks, labels, legends, annotation and drawing on subplots, matplotlib configuration

**Plotting with pandas and seaborn:** line plots, bar plots, histogram, density plots, scatter and point plots, facet grids and categorical data

## PROBABILISTIC MODELLING AND REASONING WITH PYTHON

|                         |
|-------------------------|
| Credit: 2-0-1           |
| No of Hours: 4 per week |

*Objectives: The objective of this course is to teach students the concepts of Statistics, probability, probability distribution, and other statistical methods to solve various engineering problems*

### UNIT – I

**Introduction to Statistics:** Introduction to Statistics. Role of statistics in scientific methods, current applications of statistics.

**Scientific data gathering:** Sampling techniques, scientific studies, observational studies, data management.

**Data description:** Displaying data on a single variable (graphical methods, measure of central tendency, measure of spread), displaying relationship between two or more variables, measure of association between two or more variables.

### UNIT – II

**Probability Theory:** Sample space and events, probability, axioms of probability, independent events, conditional probability, Bayes' theorem.

**Random Variables:** Discrete and continuous random variables. Probability distribution of discrete random variables, binomial distribution, poisson distribution. Probability distribution of continuous random variables, The uniform distribution, normal (gaussian) distribution, exponential distribution, gamma distribution, beta distribution, t-distribution,  $\chi^2$  distribution. Expectations, variance and covariance. Probability Inequalities. Bivariate distributions

### UNIT -III

**Point Estimations:** Methods of finding estimators, method of moments, maximum likelihood estimators, bayes estimators. Methods of evaluating estimators, mean squared error, best unbiased estimator, sufficiency and unbiasedness

**Interval Estimations:** Confidence interval of means and proportions, Distribution free confidence interval of percentiles

#### **UNIT - IV**

**Test of Statistical Hypothesis and p-values:** Tests about one mean, tests of equality of two means, test about proportions, p-values, likelihood ratio test, Bayesian tests

**Bayesian Statistics:** Bayesian inference of discrete random variable, Bayesian inference of binomial proportion, comparing Bayesian and frequentist inferences of proportion, comparing Bayesian and frequentist inferences of mean

#### **UNIT – V**

**Univariate Statistics using Python:** Mean, Mode. Median, Variance, Standard Deviation, Normal Distribution, t-distribution, interval estimation, Hypothesis Testing, Pearson correlation test, ANOVA F-test

## MACHINE LEARNING AND PATTERN RECOGNITION

|                         |
|-------------------------|
| Credit: 3-0-2           |
| No of Hours: 7 per week |

*Objectives: The objective of this course is to teach students the basic concepts of machine learning, supervised learning, unsupervised learning, and reinforcement learning*

### UNIT – I

**Introduction:** Learning systems, real world applications of machine learning, why machine learning, variable types and terminology, function approximation

**Types of machine learning:** Supervised learning, unsupervised learning, reinforcement learning

**Important concepts of machine learning:** Parametric vs non-parametric models, the trade-off between prediction accuracy and model interpretability, the curse of dimensionality, measuring the quality of fit, bias-variance trade off, overfitting, model selection, no free lunch theorem

### UNIT – II

**Linear Regression:** Linear regression, estimating the coefficients, assessing the accuracy of coefficient estimates, assessing the accuracy of the model, multiple linear regression, qualitative predictors

**Classification:** Logistic regression, estimating regression coefficients, making predictions, multiple logistic regressions, linear discriminant analysis, bayes' theorem of classification, LDA for  $p=1$ , LDA for  $p>1$ , quadratic discriminant analysis

### UNIT – III

**Resampling Methods, Model Selection and Regularization:** Cross-validation, leave-one-out cross-validation, k-fold cross-validation, the bootstrap, subset selection, shrinkage methods, ridge and lasso regression, dimension reduction methods, principal components regression, partial least square

**Tree Based Methods:** Advantages and disadvantages of trees, regression Trees, classification trees, bagging, random forest, boosting

#### **UNIT – IV**

**Support Vector Machine:** Maximum margin classifier, classification using a separating hyperplane, the maximal margin classifier, support vector classifier, support vector machines, classification with non-linear decision boundaries, support vector machine, one-versus-one classification, one-versus-many classification

#### **UNIT – V**

**Unsupervised Learning:** Principle component analysis, what are principal components, clustering methods, k-means clustering, hierarchical clustering, Independent component analysis, latent semantic indexing, Markov Models, Hidden Markov Models

## NEURAL NETWORK AND DEEP LEARNING

|                         |
|-------------------------|
| Credit: 2-0-1           |
| No of Hours: 4 per week |

*Objectives: The objective of this course is to teach students the basic concepts of neural networks, neurons, and deep learning*

### UNIT – I

**The neural network:** The neuron, linear perceptron, feed-forward neural network, limitations of linear neurons, sigmoid, tanh, relu neurons, softmax output layer, information theory, cross entropy, Kullback-Leibler divergence

**Training feed-forward neural network:** Gradient Descent, delta rules and learning rates, gradient descent with sigmoidal neurons, the backpropagation algorithms, stochastic and minibatch gradient descent, test sets, validation sets and overfitting, preventing overfitting

### UNIT – II

**TensorFlow:** Computation graphs, graphs, sessions and fetches, constructing and managing graph, flowing tensors, sessions, data types, tensor arrays and shapes, names, variables, placeholders and simple optimization, linear regression and logistic regression using tensorflow

### UNIT – III

**Implement Neural Network:** Introduction to Keras, Build neural network using Keras, Evaluating models, data preprocessing, feature engineering, feature learning, overfitting, underfitting, weight regularization, dropout, universal workflow of deep learning

### UNIT – IV

**Convolutional Neural Network:** Convolution operation, filters and feature maps, motivation, sparse interactions, parameter sharing and equivariant representation, padding and stride, max pooling, full architectural description of convolutional network, build cnn using data augmentation, using pretrained convnet, visualize what convnet learn.

## **UNIT – V**

**Models for Sequence Analysis:** Analysing Variable-length inputs, Seq2seq with neural n-gram, part of speech tagger, dependency parse, syntaxnet, recurrent neural network, challenges with vanishing gradients, long short term memory units

## **DATA ANALYTICS AND VISUALIZATION**

|                          |
|--------------------------|
| Credit: 2-0-1            |
| No of Hours : 4 Per Week |

### **UNIT - I**

**INTRODUCTION TO DATA HANDLING** Overview of Data analysis, Introduction to Data visualization, Working with statistical formulas - Logical and financial functions , Data Validation & data models, Power Map for visualize data , Power BI-Business Intelligence , Data Analysis using statistical methods, Dashboard designing.

### **UNIT - II**

**INTRODUCTION TO DATA MANIPULATION USING FUNCTION:** Heat Map, Tree Map, Smart Chart, Azure Machine learning , Column Chart, Line Chart , Pie,Bar, Area, Scatter Chart, Data Series, Axes , Chart Sheet , Trendline , Error Bars, Sparklines, Combination Chart, Gauge, Thermometer Chart , Gantt Chart , Pareto Chart etc

### **UNIT – III**

Frequency Distribution, Pivot Chart, Slicers , Tables: Structured References, Table Styles , What-If Analysis: Data Tables, Goal Seek, Quadratic Equation , Transportation Problem, Maximum Flow Problem, Sensitivity Analysis, Histogram, Descriptive, Statistics, Anova, F-Test, t-Test, Moving, Average, Exponential Smoothing | Correlation model | Regression model, Practical Lab

### **UNIT - IV**

**Data Strategy & Consumer behaviour Analytics:** Understanding Product & Category, Competitive Analysis, Market Share understanding- Market potential Index, Seasonality-Sales Trending, Consumer behaviour Analytics-MIND AND MARKET FACTORS, Budget planning & Execution- MIMI, Regression & Correlation Analysis for Sales trending, Forecasting method



with predictive investment modelling, Cohort Analysis, Google Analytics(GA), Case Studies-Assignments

## **UNIT - V**

**TABLEAU SOFTWARE: GETTING STARTED WITH TABLEAU SOFTWARE:** What is Tableau? What does the Tableau product suite comprise of? How Does Tableau Work? Tableau Architecture, What is My Tableau Repository? Connecting to Data & Introduction to data source concepts, Understanding the Tableau workspace, Dimensions and Measures, Data Types & Default Properties, Building basic views, Saving and Sharing your work-overview, Practical Lab