

R.V.R. & J.C. COLLEGE OF ENGINEERING (Autonomous)
MASTER OF COMPUTER APPLICATIONS

Course Structure, Scheme of Instruction and Examination w.e.f 2020-2021

I YEAR – I SEMESTER

Sub. Code	Subject Title	Scheme of Instruction periods per week			Scheme of Examination			
		Theory	Tutorial	Lab	Duration of Sem. End Exam. (Hrs.)	Sessional Marks	Semester End Exam. Marks	Credits
CA111	Data structures	4	1	-	3	40	60	5
CA112	Database Management Systems	4	1	-	3	40	60	5
CA113	Operating Systems	4	-	-	3	40	60	4
CA114	Computer Networks	4	-	-	3	40	60	4
CA115	Discrete Mathematical Structures	4	-	-	3	40	60	4
CA151	Data structures Lab	-	-	6	3	40	60	2
CA152	Database Management Systems Lab	-	-	6	3	40	60	2
CA153	Professional Communication Skills Lab	-	-	4	3	40	60	2
TOTAL:		20	2	16	24	320	480	28

I YEAR – II SEMESTER

Sub. Code	Subject Title	Scheme of Instruction periods per week			Scheme of Examination			
		Theory	Tutorial	Lab	Duration of Sem. End Exam. (Hrs.)	Sessional Marks	Semester End Exam. Marks	Credits
CA121	JAVA Programming	4	1	-	3	40	60	5
CA122	Web Technologies	3	1	-	3	40	60	4
CA123	Object Oriented Programming	3	1	-	3	40	60	4
CA124	Software Engineering	4	-	-	3	40	60	4
CA125	Probability and Statistics	4	-	-	3	40	60	4
CA161	Java Programming Lab	-	-	6	3	40	60	2
CA162	Web Technologies Lab	-	-	6	3	40	60	2
CA163	Python Programming Lab	-	-	6	3	40	60	2
TOTAL:		18	3	18	24	320	480	27

II YEAR – I SEMESTER

Sub. Code	Subject Title	Scheme of Instruction periods per week			Scheme of Examination			
		Theory	Tutorial	Lab	Duration of Sem. End Exam. (Hrs.)	Sessional Marks	Semester End Exam. Marks	Credits
CA211	Data Mining and Big Data	3	1	-	3	40	60	4
CA212	Artificial Intelligence	4	-	-	3	40	60	4
CA213	Cloud Computing	4	-	-	3	40	60	4
CA214	Elective-I	3	1	-	3	40	60	4
CA215	*Elective-II	3	1	-	3	40	60	4
CA251	Data Mining and Big Data Lab	-	-	6	3	40	60	2
CA252	Elective 1lab	-	-	6	3	40	60	2
CA253	Mini Project	-	-	4	-	100	-	2
CA254	Internship	-	-	-	-	100	-	2
TOTAL:		17	3	16	21	480	420	28

*Learning-By-Doing (LBD)

Elective-I	Elective-II
(A) Internet of Things (B) Machine Learning (C) Mobile Application Design and Development (D) Image Processing (E) Data Science	(A) Programming using C# (B) Devops (C) Block Chain Technology (D) Software testing and Quality Assurance (E) Design and Analysis of Algorithms

II YEAR – II SEMESTER

Sub. Code	Subject Title	Scheme of Instruction periods per week			Scheme of Examination			
		Theory	Tutorial	Lab	Duration of Sem. End Exam. (Hrs.)	Sessional Marks	Semester End Exam. Marks	Credits
CA221	MOOCs*	-	-	-	-	-	100	2
CA261	Major Project Work	-	-	-	-	40	60	10
TOTAL:		-	-	-	-	40	160	12

Note: Each student should learn any one course by registering for courses through online instruction from standard e-learning portals like SWAYAM, NPTEL, COURSERA, EdX, etc.

Enrolment of MOOCs course will be initiated from the date of commencement of class work for I Year II Semester and submit the course completion certificate on or before the last instruction day of II Year II Semester.

Three Stages in Project adjudication:

- Presentation of Problem Statement and Problem Approval by Guide.
- Progress Approval by System Demonstration with results (Internal) - 40 Marks
- Final Presentation with Documentation (External Project Viva-Voce) - 60 Marks

* MOOCs Course Completion Certificate must be submitted on or before the last instruction day of II Year II Semester, and they must clear their MOOCs.

TOTAL MARKS FOR THE MCA COURSE : 2700

TOTAL CREDITS FOR THE MCA COURSE :

I Year MCA – I Semester**CA111 : DATA STRUCTURES**

Lectures +Tutorial	: 4+1Periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 05

Prerequisite: Basic Programming and Mathematics.

Course Objectives

- To learn Algorithm complexity analysis.
- To know the structure and properties of ADT with linked lists.
- To familiarize linear and nonlinear data structures and their applications
- To understand various searching , sorting and Hashing techniques

Course Outcomes

At the end of the course the students will be able to

- Analyze the problems based on complexities.
- Implement various linear data structures and applications
- Solve problems using non-linear data structures
- Demonstrate various Sorting, Searching and Hashing techniques.

UNIT-I**15 Periods**

Introduction: Analysis of an Algorithm, Asymptotic Notations, Time and Space trade-off. Searching: Linear search and binary search techniques and their Complexity analysis.

Singly Linked lists: Representation in Memory, Abstract Data Types (ADTs) – List ADT – array-based implementation – linked list implementation —singly linked lists

UNIT-II**15 Periods**

Circularly linked lists- Doubly-linked lists – applications of lists –Polynomial Manipulation – All operations (Insertion, Deletion, Merge, Traversal).

Stacks: ADT Stack and its operations: Algorithms and their complexity analysis, Applications of stacks: Expression conversion and evaluation- corresponding algorithms and complexity and analysis

UNIT-III**15 Periods**

Queues: ADT Queue, types of Queue: Simple Queue, Circular Queue, Priority Queue; Operations on each types of Queues: Algorithm and their analysis.

Trees: Basic Tree Terminologies, Different types of trees: Binary Tree, Binary Search Tree, AVL Tree, Tree operations on each of the trees and their algorithms. Applications of Binary Trees, B-Tree definitions and algorithm

UNIT-IV**15 Periods**

Sorting Algorithms: Bubble sort – Selection sort – Insertion sort – Shell sort – Radix sort. , Heap – Applications of heap. Hashing- Hash Functions – Separate Chaining – Open Addressing – Rehashing – Extendible Hashing.

Graph Algorithms: Graphs - Data Structures for Graphs - Graph Traversals - Transitive Closure - Directed Acyclic Graphs - Shortest Paths - Minimum Spanning Trees.

Prescribed Book

1. Mark Allen Weiss, "Data Structures and Algorithm Analysis in C", 2nd Edition, Pearson Education, 1997.

Reference Books

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", Second Edition, McGraw Hill, 2002.
2. Aho, Hopcroft and Ullman, "Data Structures and Algorithms", Pearson Education, 1983.
3. Stephen G. Kochan, "Programming in C", 3rd edition, Pearson Education.
4. Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, "Fundamentals of Data Structures in C", Second Edition, University Press, 2008.
5. Reema Thareja, "Data Structures Using C", Second Edition, Oxford University Press, 2011.

Web References

1. https://www.tutorialspoint.com/data_structures_algorithms/index.htm
2. <https://www.w3schools.in/data-structures-tutorial/intro/>
3. <https://www.hackerearth.com/practice/data-structures/arrays/1-d/tutorial/>

CA112 : DATABASE MANAGEMENT SYSTEMS

Lectures +Tutorial	: 4+1Periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 05

Prerequisites: Basic Mathematics.

Course Objectives

- To understand the fundamental concepts of database system.
- To learn conceptual data modeling and relational data model.
- To know about SQL to query, update and manage a database
- To understand database design and Implementation.

Course Outcomes

At the end of the course, the students will be able to:

- Familiarize with fundamental concepts of database and various database architectures.
- Design relations for Relational databases using conceptual data modelling and implement SQL Commands.
- Identify the normalization process for relational databases.
- Understand transaction processing and concurrency control techniques for multi-user database applications.

UNIT –I**15 Periods****Databases and Database Users**

Introduction, Characteristics of the Database Approach, Actors on the Scene, Workers behind the Scene, Advantages of Using the DBMS Approach, A Brief History of Database Applications, When Not to Use a DBMS

Database System Concepts and Architecture

Data Models, Schemas, and Instances, Three-Schema Architecture and Data Independence Database Languages and Interfaces, Centralized and Client/Server Architectures for DBMSs, Classification of Database Management Systems

Data Modeling Using the Entity–Relationship (ER) Model

Using High-Level Conceptual Data Models for Database Design, Entity Types, Entity Sets, Attributes, and Keys, Relationship Types, Relationship Sets, Roles, and Structural Constraints, Weak Entity Types, Refining the ER Design for the COMPANY Database.

UNIT-II**15 Periods****The Relational Data Model and Relational Database Constraints**

Relational Model Concepts, Relational Model Constraints and Relational Database Schemas, Update Operations, Transactions, and Dealing with Constraint Violations.

Basic SQL

SQL Data Definition and Data Types, Specifying Constraints in SQL, Basic Retrieval Queries in SQL, INSERT, DELETE, and UPDATE Statements in SQL, Additional Features of SQL.

More SQL: Complex Queries, Triggers, Views, and Schema Modification

More Complex SQL Retrieval Queries, Specifying Constraints as Assertions and Actions as Triggers, Views (Virtual Tables) in SQL, Schema Change Statements in SQL.

UNIT-III**15 Periods****Basics of Functional Dependencies and Normalization for Relational databases**

Informal Design Guidelines for Relation Schemas, Functional Dependencies, Normal Forms Based on Primary Keys, General Definitions of Second and Third Normal Forms, Boyce-Codd Normal Form, Multi valued Dependency and Fourth Normal Form. Join Dependencies and Fifth Normal Form.

Relational Database Design Algorithms and Further Dependencies

Further Topics in Functional Dependencies: Inference Rules, Equivalence, and Minimal Cover, Properties of Relational Decompositions, Algorithms for Relational Database Schema Design, About Nulls, Dangling Tuples, and, Alternative Relational Designs, Further Discussion of Multivalued Dependencies and 4NF, Other Dependencies and Normal Forms.

UNIT-IV**15 Periods****Introduction to Transaction Processing Concepts and Theory**

Introduction to Transaction Processing, Transaction and System Concepts, Desirable Properties of Transactions, Characterizing Schedules Based on Recoverability , Characterizing Schedules Based on Serializability

Concurrency Control Techniques

Two-Phase Locking Techniques for Concurrency Control , Concurrency Control Based on Timestamp Ordering , Multi-version Concurrency Control Techniques , Validation (Optimistic) Techniques and Snapshot Isolation Concurrency Control , Granularity of Data Items and Multiple Granularity Locking .

Prescribed Book

1. Ramez Elmasri, Shamkant B. Navathe, "Fundamentals of Database Systems", Seventh Edition, Addison-Wesley.(chapters 1,2,3,5,6,7,14,15,20,21)

Reference Books

1. Peter Rob, Carlos Coronel, "Database Systems" - Design, Implementation and Management, Eighth Edition, Thomson (2008).
2. C.J. Date, A.Kannan, S. Swamynathan, "An Introduction to Database Systems", VII Edition Pearson Education (2006).
3. Raman A Mata - Toledo, Panline K. Cushman, "Database Management Systems", Schaum's Outlines, TMH (2007).
4. Steven Feuerstein, "Oracle PL/SQL - Programming", 10th Anniversary Edition, OREILLY (2008).

Web References

1. <http://www.nptelvideos.in/2012/11/database-management-system.html>
2. <http://www.oracle.com/technetwork/tutorials/index.html>
3. <https://www.tutorialspoint.com/dbms/>
4. <http://www.w3schools.in/dbms/>
5. <https://www.tutorialcup.com/dbms>
6. <https://www.javatpoint.com/oracle-tutorial>

7. Raman A Mata - Toledo, Panline K. Cushman, "Database Management
8. Systems", Schaum's Outlines, TMH (2007).
9. Steven Feuerstein, "Oracle PL/SQL - Programming", 10th Anniversary Edition, OREILLY (2008).

CA113 : OPERATING SYSTEMS PRINCIPLES

Lectures	: 4 Periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 04

Prerequisite: Basic knowledge in computers.

Course Objectives

- To provide coverage of basic computer system organization, overview of operating system and the concept of processes.
- To describe various features of process/thread management, CPU-scheduling and process synchronization.
- To define various deadlock handling methods and memory-management techniques.
- To understand file system interface and I/O systems.

Course Outcomes

On successful completion of the course, students should be able to:

- Explain the objectives, functions, structures of operating systems and the concept of processes.
- Understand multithreading, CPU scheduling and process synchronization.
- Categorize various deadlock handling methods, memory management techniques and the concept of mass-storage.
- Identify various file systems and I/O devices.

UNIT- I**15 Periods****Introduction**

What Operating Systems Do, Computer-System Organization, Computer-System Architecture, Operating-System Operations, Resource Management, Security and Protection, Virtualization, Distributed Systems, Computing Environments, Free and Open-Source Operating Systems

Operating-System Structures

Operating-System Services, User and Operating-System, Interface, System Calls, Operating-System Structure

Processes

Process Concept, Process Scheduling, Operations on Processes, Interprocess Communication, IPC in Shared-Memory Systems, IPC in Message-Passing Systems, Examples of IPC Systems, Communication in Client-Server Systems

UNIT – II**15 Periods****Threads & Concurrency**

Overview, Multicore Programming, Multithreading Models, Thread Libraries, Implicit Threading, Threading Issues.

CPU Scheduling

Basic Concepts, Scheduling Criteria, Scheduling Algorithms, Thread Scheduling, Multi-Processor Scheduling

Synchronization Tools & Examples

Background, The Critical-Section Problem, Peterson's Solution, Hardware Support for Synchronization, Mutex Locks, Semaphores, Monitors, Classic Problems of Synchronization

UNIT – III

15 Periods

Deadlocks

System Model, Deadlock in Multithreaded Applications, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock.

Main Memory & Virtual Memory

Background, Contiguous Memory Allocation, Paging, Structure of the Page Table, Swapping, Demand Paging, Copy-on-Write, Page Replacement, Allocation of Frames, Thrashing.

Mass-Storage Structure

Overview of Mass-Storage Structure, HDD Scheduling, NVM Scheduling, Error Detection and Correction, Storage Device Management, Swap-Space Management, Storage Attachment, RAID Structure

UNIT – IV

15 Periods

File-System Interface

File Concept, Access Methods, Directory Structure, Protection, Memory-Mapped Files

File-System Implementation

File-System Structure, File-System Operations, Directory Implementation, Allocation Methods, Free-Space Management, Efficiency and Performance

I/O Systems

Overview, I/O Hardware, Application I/O Interface, Kernel I/O Subsystem

Prescribed Book

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, "Operating System Concepts", Tenth Edition, John Wiley and Sons Inc, 2018.

Reference Books

1. William Stallings, "Operating Systems - Internals and Design Principles", Ninth Edition, Pearson Education (2017)
2. Achyut S Godbole, "Operating Systems", Third Edition, TMH (2011).
3. Ann McIverMcHoes, Ida M.Flynn, "Understanding Operating Systems", Sixth Edition, Cengage Learning (2011).
4. Deitel & Deitel, "Operating Systems", Third Edition, Pearson Education (2008).

Web Resources

1. <https://nptel.ac.in/courses/106/108/106108101/>
2. <http://www.whoishostingthis.com/resources/os-development/>

CA114 : COMPUTER NETWORKS

Lectures	: 4 Periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 04

Prerequisites: Basic knowledge in computers.

Course Objectives

- To interpret the key concepts of computer networks, reference models and transmission media.
- To establish the design issues and protocols of data link layer.
- To recognize network layer design issues, routing and transport services.
- To determine various protocols of transport layer and application layer.

Course Outcomes

At the end of the course the students will be able to

- Examine the functions and design issues of each layer in OSI and TCP/IP model.
- Explain the importance of various protocols in data link layer.
- Identify the services of network layer and transport layer.
- Understand the functions of transport layer and application layer protocols.

UNIT –I**15 Periods****Introduction:**

Uses of Computer Networks: Business Application, Home Applications, Mobile Users, Social Issues.

Network Hardware: Personal area networks, Local Area Networks, Metropolitan Area Networks, Wide Area Networks, and Internetworks.

Network Software: Protocol Hierarchies, Design Issues for the Layers, Connection Oriented versus Connectionless Services, Service Primitives, The Relationship of services to Protocols.

Reference Models: The OSI Reference Model, The TCP/IP Reference Model, A Comparison of OSI and TCP/IP reference Model, A Critique of the OSI Model and Protocols, A Critique of the TCP/IP reference model.

Physical Layer:

Guided Transmission Media: Magnetic Media, Twisted Pair, Coaxial Cable, Power Lines, and Fiber Optics.

UNIT-II**15 Periods****Data Link Layer:**

Data Link Layer Design Issues: Services Provided to the Network Layer, Framing, Error Control, and Flow Control.

Error Detection and Correction: Error correcting Codes, Error Detecting Codes.

Elementary Data Link Protocols: A Simplex Protocol, A simplex Stop- and – wait Protocol for an error-free channel, a simplex stop-and wait protocol for a noisy channel.

Sliding Window Protocols: A one-bit sliding Window Protocol, A Protocol using Go-Back-N, A Protocol using Selective Repeat.

Ethernet: Classic Ethernet physical layer, Classic Ethernet MAC Sublayer protocol, Switched Ethernet, Fast Ethernet, Gigabit Ethernet.

Data Link Layer Switching: Uses of bridges, Learning Bridges, Spanning Tree bridges, Repeaters, hubs, bridges, switches, routers and gateways.

UNIT – III

15 Periods

The Network Layer:

Network Layer Design Issues: Store and Forward Packet Switching, Services provided to the Transport Layer, Implementation of Connectionless Services, Implementation of Connection Oriented Services, Comparison of Virtual Circuit and Datagram subnets.

Routing Algorithms: The Optimality Principle, Shortest Path Routing, Flooding, Distance Vector Routing, Link State Routing, Hierarchical Routing ,Broadcast Routing, Multicast Routing.

The Transport Layer:

The Transport Service: Services provided to the Upper Layers, Transport Services Primitives.

Elements of Transport Protocols: Addressing, Connection Establishment, Connection Release, Flow Control and Buffering, Multiplexing, Crash Recovery.

UNIT – IV

15 Periods

The Internet Transport Protocols: UDP- Introduction to UDP, Remote Procedure Call, and The Real-time Transport Protocol.

The Internet Transport Protocols: TCP- Introduction to TCP, The TCP Service Model, The TCP Protocol, The TCP segment header, TCP connection establishment, TCP connection release – TCP connection management Modeling.

The Application Layer:

DNS-The Domain Name System: The DNS Name Space, Domain Resource Records, Name Servers.

Electronic Mail: Architecture and Services, the User Agent, Message Formats, Message Transfer, Final Delivery.

The World Wide Web: Architectural Overview, Static Web Pages, HTTP – The Hyper Text Transfer Protocol.

Prescribed Book

1. Andrew S. Tanenbaum, David S. Wetherall “Computer Networks”, Fifth Edition, Pearson.

Reference Books

1. James F.Kurose, Keith W.Ross, “Computer Networking”, Third Edition, Pearson.
2. Behrouz A Forouzan, “Data Communications and Networking”, Fourth Edition, TMH.
3. Michael A. Gallo, William M. Hancock, “Computer Communications and Networking Technologies”, Cengage Learning (2008).

Web References

1. https://wps.pearsoned.com/ecs_kurose_compnetw_6/
2. <https://www.pearsonhighered.com/cs-resources/products/series.html>

CA115 : DISCRETE MATHEMATICAL STRUCTURES

Lectures	: 4 Periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 04

Prerequisite: Mathematics background up to XII class.

Course Objectives

At the end of the course the students will

- Understand and construct mathematical arguments.
- Develop recursive algorithms based on mathematical induction and Know basic properties of relations.
- Know essential concepts of algorithms.
- Know essential concepts in trees and Graphs.

Course Outcomes

At the end of the course the students will be able to familiar with

- Mathematical Logic.
- Relations and recursion.
- Techniques of counting.
- Graphs and trees, relations.

UNIT – I

15 Periods

The Foundations: Logic and Proofs: Propositional Logic, Applications of Propositional Logic, Propositional Equivalences, Predicates and Quantifiers , Nested Quantifiers, Rules of Inference, Introduction to proofs, Proof Methods and Strategy.
Basic Structures: Sets, Functions, Sequences, Sums, and Matrices: Sets, Set Operations, Functions, Sequences and Summations.

UNIT – II

15 Periods

Induction and Recursion: Mathematical Induction, Strong Induction and Well-Ordering, Recursive Definitions and Structural Induction, Recursive Algorithms.
Relations: Relations and Their Properties, n-ary Relations and Their Applications, Representing Relations, Closures of Relations, Equivalence Relations, Partial Orderings.

UNIT – III

15 Periods

Counting: The Basics of Counting, The Pigeon Hole Principle, Permutations and Combinations, Binomial Coefficients, Generalized Permutations and Combinations.
Advanced Counting Techniques: Applications of Recurrence Relations, Solving Linear Recurrence Relations, Divide-and-Conquer Algorithms and Recurrence Relations, Generating Functions, Inclusion–Exclusion, Applications of Inclusion–Exclusion.

UNIT – IV

15 Periods

Graphs: Graphs and Graph Models, Graph Terminology and Special Types of Graphs, Representing Graphs and Graph Isomorphism's, Connectivity, Euler and Hamilton Paths, Shortest Path Problems, Planar Graphs, Graph Coloring.

Trees: Introduction to Trees, Applications of Trees, Tree Traversal, Spanning Trees, Minimum Spanning Trees.

Prescribed Book

1. Kenneth Rosen, "Discrete Mathematics and Its Applications", 7/e, McGraw Hill Publishing Co., 2012.

Reference Books

1. Joe L. Mott, Abraham Kandel & T. P. Baker, "Discrete Mathematics for computerscientists & mathematicians", 2nd edition, Prentice Hall of India Ltd, New Delhi (2009).
2. Swapan Kumar Sarkar, "A Text Book of Discrete Mathematics", 5th edition S.Chand (2013).
3. D.S.Malik and M.K.Sen, "Discrete Mathematical Structures", 1st edition, Cengage (2012).

Web References

1. <http://nptel.ac.in/courses/106106094/>
2. <http://csvls.blogspot.in/2010/04/video-lectures-of-discrete-mathematics.html>

CA151 : DATA STRUCTURES LAB

Lectures	: ---	Sessional Marks	: 40
Practicals	: 6 Periods/week	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 02

Prerequisite: Basic Programming and Mathematics.

Course Objectives

At the end of the course the students will learn

- skills to design and analyze simple linear data structures
- skills to design and analyze simple nonlinear data structures
- the ability to identify and apply the suitable data structure for the given real world problem
- To Gain knowledge in practical applications of data structures

Course Outcomes

At the end of the course the students will be able to

- design and analyze the time and space efficiency of the data structure
- identify the appropriate data structure for given problem
- design and analyze simple linear data structures
- solve problems using nonlinear data structures

CYCLE - I

1. Implement the following operations on single linked list.
2. Implement the following operations on double linked list.
3. The Josephus problem is the following game: N people, numbered 1 to N, are sitting in a circle. Starting at person 1, a hot potato is passed. After M passes, the person holding the hot potato is eliminated, the circle closes ranks, and the game continues with the person who was sitting after the eliminated person picking up the hot potato. The last remaining person wins. Thus, if $M = 0$ and $N = 5$, players are eliminated in order, and player 5 wins. If $M = 1$ and $N = 5$, the order of elimination is 2, 4, 1, 5. Write a program to solve the Josephus problem for general values of M and N.
4. Merge two given linked lists.
5. Perform Addition and multiplications on polynomials.

CYCLE - II

6. Implement Stack Operations Using Arrays.
7. Implement Stack Operations Using Linked List.
8. Implement Queue Operations Using Arrays.
9. Implement Operations on Queue using circularly linked list.
10. Implement Operations on double ended Queue.
11. Converting infix expression to postfix expression by using stack.
12. Write program to evaluate post fix expression.

CYCLE – III

13. Implement binary search technique recursively.
14. Sort given elements using
 - a. Selection Sort.

- b. Insertion Sort.
- c. Merge Sort.
- d. Quick Sort.
- e. Heap Sort

15. Construct BST and implement traversing techniques recursively.

Web References:

1. https://www.tutorialspoint.com/data_structures_algorithms/index.htm
2. <https://www.w3schools.in/data-structures-tutorial/intro/>
3. <https://www.hackerearth.com/practice/data-structures/arrays/1-d/tutorial/>

CA152 : DATABASE MANAGEMENT SYSTEMS LAB

Lectures	: ---	Sessional Marks	: 40
Practicals	: 6 Periods/week	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 02

Prerequisite: Basic Mathematics.

Course Objectives

At the end of the course the students will understand

- The concepts of relational model and its applications
- Designing databases.
- DDL,DML,DCL commands
- PL/SQL Programming.

Course Outcomes

At the end of the course the students will be able to

- Understand how to create and place constraints on databases.
- Write simple queries to retrieve and manipulate data.
- Summarize data by means of group by operation and arranging the records using order by operation
- Write PL/SQL programs for small applications

CYCLE – I

Aim:Marketing Company wishes to computerize their operations by using following tables.

Table Name: Client_Master

Description:This table stores the information about the clients.

Column Name	Data Type	Size	Attribute
Client_no	Varchar2	6	Primary Key and first letter should starts with 'C'
Name	Varchar2	10	Not null
Address1	Varchar2	10	
Address2	Varchar2	10	
City	Varchar2	10	
State	Varchar2	10	
Pincode	Number	6	Not null
Bal_due	Number	10,2	

TableName: Product_master

Description: This table stores the information about products.

Column Name	Data Type	Size	Attribute
Product_no	Varchar2	6	Primary Key and first letter should starts with 'P'
Description	Varchar2	10	Not null
Unit_measure	Varchar2	10	
Qty_on_hand	Number	8	
Record_ivl	Number	8	
Sell_price	Number	8,2	Not null, can't be 0
Cost_price	Number	8,2	Not null, can't be 0

TableName:salesman_master

Description:This table stores the salesmen working in the company

Column Name	Data Type	Size	Attribute
Salesman_id	Varchar2	6	Primary Key and first letter should starts with 'S'
Name	Varchar2	10	Not null
Address1	Varchar2	10	
Address2	Varchar2	10	
City	Varchar2	10	
State	Varchar2	10	
Pincode	Number	6	Not null
Sal_amt	Number	8,2	Should not null and zero
Target_amt	Number	8,2	Should not null and zero
Remarks	Varchar2	10	

TableName: sales_order

Description: This table stores the information about orders

Column Name	Data Type	Size	Attribute
S_order_no	Varchar2	6	Primary Key and first char is 'O'
S_order_date	Date		
Client_no	Varchar2	6	Foreign key
Delve_address	Varchar2	20	
Salesman_no	Varchar2	6	Foreign key
Delve_type	Varchar2	1	Delivery: part(P)/Full(F) and default 'F'
Billed_yn	Char	1	
Delve_date	Date		Can't be less than the s_order_date
Order_status	Varchar2	10	Values in 'IN PROCESS', 'FULFILLED', 'BACK ORDER', 'CANCELLED'
Remarks	Varchar2	10	

TableName:sales_order_details

Description:This table stores the information about products ordered

Column Name	Data Type	Size	Attribute
S_order_no	Varchar2	6	Primary key, foreign key references sales_order table
Product_no	Varchar2	6	Primary key, foreign key references product_master table
Qty_ordered	Number	8	
Qty_disp	Number	8	
Total_amt	Number	10,2	

TableName:challan_master

Description:This table stores the information about challansmade for orders.

Column Name	Data Type	Size	Attribute
Challan_no	Varchar2	6	Primary key, first two letters must start with 'CH'
S_order_no	Varchar2	6	Foreign key references sales_order
Challan_date	Date		

Billed_yn	Char	1	Values in 'Y', 'N' default 'N'
-----------	------	---	--------------------------------

TableName:Challan_Details

Description:This table stores the information about challan details.

Column Name	Data Type	Size	Attribute
Challan_no	Varchar2	6	Primary key, foreign key references challan_master table
Product_no	Varchar2	6	Primary key, foreign key references product_master table
Qty_disp	Number	8	Not null

Solve the following queries by using above tables

1. Retrieve the list of names and cities of all the clients.
2. List the various products available from product_master.
3. Find out the clients who stay in a city whose second letter is 'a'.
4. Find the list of all clients who stay in the city 'CHENNAI' or 'DELHI'.
5. List all the clients located at 'CHENNAI'.
6. Print the information from sales order as the order the places in the month of January.
7. Find the products with description as 'Floppy Drive' and 'Pen drive'.
8. Find the products whose selling price is greater than 2000 and less than or equal to 5000.
9. Find the products whose selling price is more than 1500 and also find the new selling price as original selling price *15.
10. Find the products in the sorted order of their description.
11. Divide the cost of product '540 HDD' by difference between its price and 100.
12. List the product number, description, sell price of products whose description begin with letter 'M'.
13. List all the orders that were cancelled in the month of March.
14. Count the total number of orders.
15. Calculate the average price of all the products.
16. Determine the maximum and minimum product prices.
17. Count the number of products having price grater than or equal to 1500.
18. Find all the products whose quantity on hand is less than reorder level.
19. Find out the challan details whose quantity dispatch is high.
20. Find out the order status of the sales order, whose order delivery is maximum in the month of March.
21. Find out the total salesmade by the each salesman.
22. Find the total revenue gained by the each product sales in the period of Q1 and Q2 of year 2006.
23. Print the description and total qty sold for each product..
24. Find the value of each product sold.
25. Calculate the average qty sold for each client that has a maximum order value of 1,50,000.

CYCLE - II (Employee Database)

Aim:An enterprise wishes to maintain a database to automate its operations. Enterprise divided into to certain departments and each department consists of employees. The following two tables describes the automation schemas

DEPT (DEPTNO, DNAME, LOC)

EMP (EMPNO,ENAME,JOB,MGR,HIREDATE,SAL,COMM,DEPTNO)

1. Create a view, which contain employee names and their manager names working in sales department.
2. Determine the names of employee, who earn more than there managers.
3. Determine the names of employees, who take highest salary in their departments.
4. Determine the employees, who located at the same place.
5. Determine the employees, whose total salary is like the minimum salary of any department.
6. Update the employee salary by 25%, whose experience is greater than 10 years.
7. Delete the employees, who completed 32 years of service.
8. Determine the minimum salary of an employee and his details, who join on the same date.
9. Determine the count of employees, who are taking commission and not taking commission.
10. Determine the department does not contain any employees.
11. Find out the details of top 5 earners of company. (Note: Employee Salaries should not be duplicate like 5k,4k,4k,3k,2k)
12. Display managers name whose salary is more than an average salary of his employees.
13. Display the names of the managers who is having maximum number of employees working under him?
14. In which year did most people join the company? Display the year and number of employees.
15. Display ename, dname even if there no employees working in a particular department(use outer join).

PL/SQL PROGRAMS

1. The hrd manager has decided to raise the employee salary by 15%. Write a pl/sql block to accept the employee number and update the salary of that employee. Display appropriate message based on the existence of the record in emp table.
2. Write a pl/sql program to display top 10 rows in emp table based on their job and salary.
3. Write a pl/sql program to raise the employee salary by 10%, for department number 30 people and also maintain the raised details in the raise table.
4. Write a pl/sql procedure to prepare an electricity bill by using following table used:
elect

NAME	NULL?	TYPE
MNO	NOT NULL	NUMBER(3)
CNAME		VARCHAR2(20)
CUR_READ		NUMBER(5)
PREV_READ		NUMBER(5)

NO_UNITS	NUMBER(5)
AMOUNT	NUMBER(8,2)
SER_TAX	NUMBER(8,2)
NET_AMT	NUMBER(9,2)

5. Write a pl/sql procedure to evaluate the grade of a student with following conditions:

for pass: all marks > 40

for i class: total%>59

for ii class: total%between>40 and <60

for iii class: total%=40

Also maintain the details in abstract table.

Tables USED

1. TABLE STD

NAME	NULL?	TYPE
-----	-----	-----
NO	NOT NULL	NUMBER
NAME		VARCHAR2(10)
INTNO		NUMBER
CLASS	NOT NULL	VARCHAR2(10)
M1		NUMBER
M2		NUMBER
M3		NUMBER
M4		NUMBER
M5		NUMBER

2. TABLE ABSTRACT

NAME	NULL?	TYPE
-----	-----	-----
STDNO		NUMBER
STDNAME		VARCHAR2(10)
CLASS		VARCHAR2(10)
MONTH		VARCHAR2(10)
INTNO (INTERNAL NUMBER)		NUMBER
TOT		NUMBER
GRADE		VARCHAR2(10)
PERCENT		NUMBER
DAT_ENTER		DATE

6. Create an object to describe the details of address type data.
7. Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers.

Reference Books

1. Kevin Loney, Oracle Database 10g The Complete Reference. Tata McGraw-Hill Publishing Company Limited.

2. Scott Urman, Oracle 9i PL/SQL Programming. Tata McGraw-Hill Publishing Company Limited.
3. Parteek Bhatia, SanjivDatta and Ranjit Singh, Simplified Approach to Oracle. Kalyani Publishers.

Web References

1. <https://www.tutorialspoint.com/dbms/>
2. <http://www.studytonight.com/dbms/introduction-to-sql>
3. <http://nptel.ac.in/courses/105102015/37>
4. <https://www.javatpoint.com/sql-tutorial>

CA153: PROFESSIONAL COMMUNICATION SKILLS LAB

Lectures	: ---	Sessional Marks	: 40
Practical	: 4 Periods/week	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 02

Prerequisite: Basic Knowledge in English

Course Objectives:

- To enable students, demonstrate an understanding of a grammar structure by talking about it in pair and group work, and class Discussions.
- To help Students, demonstrate consistent and appropriate language use in extended conversations and discussions.
- To develop the art of effective reading and answer comprehension passages.
- To equip with appropriate and spontaneous speech dynamics.

Course Outcomes:

After successful completion of the course, the students will be able to:

- Demonstrate an understanding of a grammar structure by talking about it in pair and group work, and class Discussions.
- Establish consistent and appropriate language use in extended conversations and discussions.
- Use different skills, inferring lexical and contextual meaning and attempt comprehension passages.
- Develop appropriate speech dynamics in professional situations.

UNIT - I**Functional Grammar**

- Articles & Prepositions
- Parts of Speech
- Tenses & Voice
- Direct and Indirect Speech
- Question tags
- Correction of Sentences

UNIT - II**Communication Skills**

- Expressions in Various Situations
- Contextual Vocabulary, Phrases & Native Expressions
- Greeting and Introducing
 - Making Requests
 - Agreeing and Disagreeing
 - Asking for and Giving Permissions; Offering Help
 - Telephonic Etiquette
 - Non-verbal Cues in Communication

UNIT – III

Reading Comprehension

- The Art of Effective Reading – Benefits of Effective Reading
- Types & Methods of Reading
- Different Passages for Reading Comprehension
- Identifying the Central Idea – Inferring Lexical and Contextual Meaning.

UNIT – IV

Interactive Classroom Activities

- Dynamics of Professional Presentations – Individual & Group Presentations
- Delivering Just-a-minute (JAM) Sessions, Debate.
- Describing People, Places, Events & Things
- Role play

Reference books

1. Kiranmai Dutt, Rajeevan, C.L.N Prakash, A course in English Communication 2013.
2. Communication Skills - Sanjay Kumar & Pushpa Latha (OUP)- 2nd Impression, 2012
3. Technical Communication - Meenakshi Raman & Sangeeta Sharma, Oxford Semester Press, 6th Impression, 2012
4. Business Communication, II Ed, OUP, by Meenakshi Raman & Prakash Singh, 2012.

Software

1. Pronunciation power I&II
2. Author Plus
3. Call Centre Communication – Clarity

Web References

1. <https://www.indiabix.com/verbal-reasoning/questions-and-answers/>
2. <https://writeandimprove.com/>
3. <http://guidetogrammar.org/grammar/>
4. <https://learnenglish.britishcouncil.org/skills/listening>
5. <https://ielts-up.com/ielts-speaking-simulator.html>

I Year MCA – II Semester**CA121: JAVA PROGRAMMING**

Lectures +Tutorial	: 4+1Periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 05

Prerequisite: CA111(Data Structures), CA151(Data Structures Lab).

Course Objectives

The main objectives of this course are:

- To learn the basics of java concepts and fundamentals of platform independent object oriented language.
- To understand the concept as well as the purpose and usage principles of inheritance, polymorphism, interfaces and packages.
- To develop skills in writing programs using exception handling techniques and multithreading.
- To understand the Event Handling, Applets and Networking and to gain in-depth understanding of Java Beans and Servlets.

Course Outcomes

At the end of the course, student will be able to

- Apply the syntax and semantics of java programming language and basic principles of OOPs.
- Understand the concepts of Packages, String, Exception handling and Multithreading.
- Demonstrate how the java program communicates with the console and disk files using the concept of streams and also Implements Applets and Event Handling.
- Demonstrate Network, Java Beans and server side applications.

UNIT – I**15 Periods**

The History and Evolution of Java, An Overview of Java, Data Types, Variables, and Arrays, Operators, Control Statements.

Introducing Classes: Class fundamentals, Declaring the objects, Assigning Object Reference Variables, Introducing Methods, Constructors, The this keyword, Garbage Collection, the finalize() Method.

A Closer Look at Methods and Classes: Overloading Methods, Using objects as Parameters, Returning Objects, Introducing Access control, Understanding static and final keywords, Nested and Inner Classes.

Inheritance: Inheritance Basics, Using super, creating multilevel Hierarchy, when Constructors are executed, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, using final with Inheritance, The Object class.

UNIT – II**15 Periods**

Packages and Interfaces: Packages, Access Protection, Importing Packages, Interfaces, Default Interface Methods, Use static Methods in an Interface.

Strings: Exploring the String class, String buffer class, Command-line arguments.

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

Multithreading: The Java Threaded Model, The Main Thread, Creating a Thread, Creating Multiple Threads, Using `isAlive()` and `join()`, Thread Priorities, Synchronization, Inter Thread Communication, Suspending, Resuming, Stopping Threads, Obtaining A Thread's State.

UNIT – III**15 Periods**

I/O Basics: Streams, Byte streams, Character streams, Reading Console Input, Writing Console Output, The `PrintWriter` Class, Reading and Writing Files.

Applets - Applet Fundamentals, The Applet Class: Applet Basics, Applet Architecture, An Applet Skeleton, Simple Applet Display Methods, Requesting Repainting, The HTML APPLET Tag, Passing Parameters to Applets.

Event Handling: Two Event Handling Mechanisms, The Delegation Event Model, Event Classes, The `KeyEvent` Class, Sources of Events, Event Listener Interfaces, Using The Delegation Event Model, Adapter Classes , Inner Classes.

UNIT – IV**15 Periods**

Networking: Networking Basics, The Networking Classes and Interfaces, `InetAddress`, TCP/IP Client sockets, URL, URL connection, TCP/IP sockets Server Sockets, Datagrams.

Java Beans: What is a Java Bean? Advantages of Java Bean, Introspection, Bound and Constrained Properties, Persistence, Customizers, The Java Bean API, A Bean Example.

Introducing Servlets: Background, Lifecycle of a Servlet, Servlet development options, The ServletAPI, The `javax.servlet` Package, Reading Servlet parameters, The `javax.servlet` HTTP package, Handling Http Request & Responses, Using Cookies, Session Tracking.

Prescribed Book

1. Herbert Schildt, "Java The Complete Reference", 9th Edition, McGraw Hill Education (India) Private Limited, New Delhi.

Reference Books

1. Paul Dietel and Harvey Dietel, "Java: How to Program", Ninth Edition, PHI
2. Cay Horstmann, "Big Java", 4th Edition, JOHN WILEY & SONS, INC.
3. Y. Daniel Liang, "Introduction to Java programming", Pearson Publication.

Web References

1. <http://www.cplusplus.com/reference/>
2. <http://en.cppreference.com/w/>
3. <http://www.decompile.com/>
4. <http://www.programmingsimplified.com/cpp>
5. <http://www.learncpp.com/>
6. <http://www.stroustrup.com/>

CA122: WEB TECHNOLOGIES

Lectures +Tutorial	: 3+1Periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 04

Prerequisite: CA114(Computer Networks)..

Course Objectives

- To explain basic technologies to develop web documents.
- To apply CSS to web documents.
- To demonstrate Dynamic HTML Pages and Event handling mechanism.
- To illustrate XML, Web Servers and JDBC and to describe PHP technologies.

Course Outcomes

At the end of this course students will be able to

- Create static web pages using HTML, CSS and Evaluate client side Web pages using JavaScript.
- Design and describe Dynamic Web pages using JavaScript event handling.
- Create XML documents and work with web servers to create web applications with JDBC.
- Design and Develop server side programs using PHP Technologies.

UNIT-I**15 Periods**

Introduction to HTML 5: Part 1 and Part 2.

Introduction to Cascading Style Sheets (CSS): Part 1 and Part 2.

JavaScript: Introduction to Scripting, Control Statements Part - I & II

UNIT-II**15 Periods**

JavaScript: Functions, Arrays, Objects.

DOM: Objects and Collections

JavaScript Event Handling: A deeper look.

HTML 5: Introduction to canvas: Introduction- canvas Coordinate SystemRectangles- Using Paths to Draw Lines- Drawing Arcs and CirclesShadows-Quadratic Curves- Linear Gradients- Images-Image Manipulation: Processing the Individual Pixels of a canvas-Patterns- TransformationsText-Resizing the canvas to fill the Browser Window

UNIT-III**15 Periods**

XML: Introduction- XML Basics- Structuring Data-XML NamespacesDocument Type Definitions (DTDs)- W3C XML Schema Documents-.XML Vocabularies- Extensible Stylesheet Language and XSL Transformations, Document Object Model (DOM).

JDBC: Introduction to JDBC - Connections - Internal Database Connections - Statements - Results Sets - Prepared Statements - Callable Statements.

UNIT-IV**15 Periods**

Web Servers: Introduction-HTTP Transactions- Multitier Application Architecture- Client-Side Scripting versus Server-Side Scripting, Accessing Web Servers.

PHP: Introduction, Simple PHP Program, Converting Between Data Types, Arithmetic Operators, Initializing and Manipulating Arrays, String Comparisons, String Processing

with Regular Expressions, Form Processing and Business Logic, Reading from a Database, Using Cookies, Dynamic Content.

Prescribed Books

1. Paul Deitel, Harvey.Deitel, Abbey.Deitel, "Internet and World Wide Web HowTo Program", Fifth Edition, PHI Pvt.Ltd.
2. Donald Bales, "Java Programming with JDBC", O'Reilly.

Reference Books

1. "Web Technologies – Black Book", drematech, 2013.
2. Jason Cranford Teague "Visual Quick Start Guide CSS, DHTML & AJAX", 4th edition, Pearson Education.
3. Tom Nerino Doli Smith "JavaScript & AJAX for the Web" Pearson Education, 2007.
4. Uttam K. Roy, "Web Technology", Oxford University Press, 2010.

Web References

1. www.w3schools.com
2. www.tutorialspot.com
3. www.deitel.com

CA123: OBJECT ORIENTED PROGRAMMING

Lectures +Tutorial	: 3+1Periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 04

Prerequisite: Basic Mathematics and Programming.

Course Objectives

The objectives of the course are to:

- Introduce the C++ fundamentals and Object oriented features in C++
- Introduce fundamentals of Python Programming language.
- Teach students Lists, defining functions in Python.
- Demonstrate object orientation concepts , mutable immutable types in Python

Course Outcomes

At the end of the course the students will be able to

- Solve the problems using C++
- Explain the fundamentals of Python programming language.
- Solve the problems in Python by defining modules,files, exception handling.
- Compare and contrast mutable and immutable datatypes and can demonstrate object orientation in Python in solving real world problems

UNIT – I**15 Periods**

Principles of Object-Oriented Programming, Beginning with C++, Tokens, Expressions and Control Structures, Functions in C++, Classes and Objects, Constructors and Destructors, Operator Overloading and Type conversions

UNIT – II**15 Periods**

Inheritance Extending Classes, Pointers, Virtual Functions and Polymorphism
Data and Expressions - Literals, Variables and Identifiers, Operators, Expressions and Data Types.
Control Structures - What Is a Control Structure? Boolean Expressions (Conditions), Selection Control, Iterative Control.

UNIT – III**15 Periods**

Lists - List Structures, Lists (Sequences) in Python, Iterating Over Lists (Sequences) in Python, More on Python Lists.
Functions - Program Routines, More on Functions.
Objects and Their Use - Software Objects, Turtle Graphics
Modular Design – Modules, Python Modules.
Text Files - What Is a Text File? Using Text Files, String Processing, Exception Handling

UNIT – IV**15 Periods**

Dictionaries and Sets - Dictionary Type in Python, Set Data Type.
Object-Oriented Programming - What Is Object-Oriented Programming?
Encapsulation, Inheritance, Polymorphism.
Recursion - Recursive Functions, Recursive Problem Solving, Iteration vs. Recursion.

Prescribed Books

1. Object Oriented Programming with C++ (Sixth Edition) by E. Balagurusamy, Tata McGraw-Hill Education Pvt. Ltd
2. Introduction to Computer Science Using Python: A Computational Problem-Solving Focus by Charles Dierbach, Wiley.

Reference Books

1. The Complete Reference - C++ by Herbert Schildt, 4/e, Tata McGraw Hill.
2. The C++ Programming Language by Bjarne Stroustrup, Special Edition, Pearson india
3. Fundamentals of Python Programming : Richard L. Halterman by Southern Adventist University
4. Think Python: How to Think Like a Computer Scientist, Allen Downey, Green Tea Press, Version 2.0.17

Web References

1. <https://www.w3schools.com/cpp/>
2. <https://www.w3schools.com/python/>
3. <https://www.tutorialspoint.com/cplusplus/index.htm>
4. <https://www.tutorialspoint.com/python/index.htm>
5. https://onlinecourses.nptel.ac.in/noc20_cs70/preview

CA124 : SOFTWARE ENGINEERING

Lectures	: 4 Periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 04

Prerequisites: Problem solving Techniques.

Course Objectives

At the end of the course, the student will understand

- The basic concepts on Software Engineering Methods and Process Models.
- The agile software development with a comprehensive set of skills appropriate to the needs of the dynamic global computing-based society.
- How requirements are Modelling and design.
- Quality management and how to ensure good quality software by applying various Testing Strategies.

Course outcomes

At the end of the course, the student will be able to

- Apply the software engineering lifecycle.
- Analyze and specify software requirements.
- Design, develop and deploy software projects.
- Evaluate and assess the quality of the software.

UNIT - I**15 Periods**

Software and Software Engineering: The nature of Software, Software Engineering, The Software Process, Software Myths.

The software Process: Process models, Prescriptive Process Models: The Waterfall Model, Incremental Process Models, Evolutionary Process Models, Concurrent Models.

Specialized Process models: Component based Development, The Formal Methods Model, Aspect Oriented Software Development.

The Unified Process: Phases of the Unified Process.

UNIT- II**15 Periods**

Agile Development: What Is Agility? What Is an Agile Process? Agile process models: Adaptive Software Development, Extreme Programming, Scrum, Dynamic Systems Development Method, Crystal, Feature driven Development, Lean Software Development, Agile Modeling.

Understanding Requirements: Requirements Engineering, Establishing the Groundwork, Eliciting requirements, Developing Use Cases, Building the requirements Model.

UNIT - III**15 Periods**

Requirements Modelling: Scenarios, Information, and Analysis Classes:

Requirement Analysis, Scenario-based Modelling, UML Models That Supplement the Use Case, Data Modelling Concepts, Class Based Modelling.

Design Concepts: Design within the Context of Software Engineering, The Design Process, Design Concepts, **The Design Model:** Data Design Elements, Architectural Design Elements, Interface Design Elements, Component-Level Design Elements.

UNIT - IV**15 Periods****Quality Management:** What is Quality?, Achieving Software Quality,**Review Techniques:** Cost Impact of Software Reviews, Defect amplification and removal, Informal and Formal Reviews, **Software Quality Assurance:** Elements of SQA, Software Reliability.**Software Testing Strategies:** A Strategic Approach to Software Testing, Test Strategies for Conventional Software, Validation Testing, System Testing, The Art of Debugging.**Testing Conventional Applications:** Software testing Fundamentals, Internal and External Views of Testing, White-Box Testing, Basis Path Testing, Control Structure Testing, Black-Box Testing, Model-Based Testing.**Prescribed Book**

1. Roger S. Pressman, Software Engineering - A Practitioner's Approach, Seventh Edition, McGraw Hill Publications.

Reference Books

1. Ugrasen Suman, Software Engineering, Concepts and Practices, Cengage Publications,
2. Ian Sommerville, Software Engineering, Sixth Edition, Pearson Education.
3. Agile Project Management: Best Practices and Methodologies - <https://www.altexsoft.com/whitepapers/agile-project-management-best-practices-and-methodologies/>
4. Carlo Ghezzi, Mehdi Jazayeri, Dino Mandrioli, Fundamentals of Software Engineering, Second Edition, PHI.
5. Rajib Mall, Fundamentals of Software Engineering, Second Edition, PHI.
6. Software Engineering Resources : - www.rspa.com/spi/
7. Carnegie Mellon University, Software Engineering Institute, www.sei.cmu.edu/

Web References

1. <https://archive.org/details/SoftwareEngineering7thEDByRogerS.Pressman>
2. <http://ceit.aut.ac.ir/~91131079/SE2/SE2%20Website/Lecture%20Slides.html>
3. www.rspa.com/spi/www.sei.cmu.edu/
4. <http://www.pearsonhighered.com/educator/product/Software-Engineering/9780137035151.page>
5. <http://www.agilemanifesto.org/>
6. http://www.tutorialspoint.com/software_engineering/
7. <http://www.tutorialride.com/software-engineering/software-engineering-tutorial.htm>

CA125 : PROBABILITY & STATISTICS

Lectures	: 4 Periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 04

Prerequisite: Basic Mathematics.

COURSE OBJECTIVES

To provide knowledge on basic probability distributions and densities and their applications.

- To provide skills in applying the basic principles of generalizations from sample data.
- To provide skills in applying statistical inference to practical problems for the sample data.
- To provide the knowledge of curve fitting, correlation and regression analysis to the data and estimate the future values to draw the conclusions

COURSE OUTCOMES

The student who successfully completes this course will be able to:

- Understand the basic principles in distribution theory, mathematical expectation, and various application of probability distributions.
- Understand the basic principles of generalizations from sample data
- Understand the basic principles of statistical inference (i.e. estimation and hypothesis testing).
- Develop the linear and non linear regression models and find the relationship between the variables under study and use them for estimating the future values.

UNIT- I (DISTRIBUTION THEORY)**15 Periods**

Probability Distributions: Random Variables, Binomial distribution, Hypergeometric distribution, Mean and Variance of a Probability distribution, Chebyshev's theorem, Poisson Approximation to the Binomial distribution.

Probability Densities: Continuous Random Variables, Normal Distribution, Normal Approximation to the Binomial Distribution, other Probability Densities: Uniform Distribution, Log-Normal Distribution, Gamma Distribution, Beta Distribution, Weibull distribution.

(Sections 4.1 to 4.6 and 5.1 to 5.9 of the Text Book)

UNIT - II (TESTING OF HYPOTHESIS)**15 Periods**

Sampling Distribution: Populations and samples, The sampling distribution of the mean (σ known), The sampling distribution of the mean (σ unknown), The sampling distribution of the variance.

Inferences Concerning Means: Point Estimation, Interval Estimation, Tests of Hypotheses, Null Hypotheses and Tests of hypotheses, Hypotheses Concerning One Mean, Relation between Tests and Confidence Intervals, Operating Characteristic Curves, Inferences Concerning Two Means.

(Sections 6.1 to 6.4 and 7.1 to 7.8 of the Text Book)

UNIT - III (TESTING OF HYPOTHESIS contd...)

15 Periods

Inferences Concerning Variances: Estimation of Variances, Hypotheses Concerning One Variance, Hypotheses Concerning Two Variances.

Inferences Concerning Proportions: Estimation of Proportions, Hypotheses Concerning One Proportion, Hypotheses Concerning Several Proportions, The analysis of $r \times c$ tables, Goodness of fit

(Sections 8.1 to 8.3 and 9.1 to 9.5 of the text book)

UNIT - IV (CURVE FITTING , CORRELATION and REGRESSION)

15 Periods

Curve Fitting: The method of Least squares, Inferences based on Least squares estimators.

Correlation and Regression: Curvilinear regression, Multiple regression, Checking the adequacy of the model, Correlation, Multiple linear regression.

(Sections 11.1 to 11.7 of the text book)

Prescribed Book

1. Miller & Freund's: "Probability and Statistics for Engineers", Sixth Edition, PHI by Richard A. Johnson.

Reference Book

1. J Susan Milton and Jesse C. Arnold: "Introduction to Probability and Statistics", Fourth edition, TMH,(2007).

CA161 : JAVA PROGRAMMING LAB

Lectures	:---	Sessional Marks	: 40
Practicals	: 6 Periods/week	Sem.End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 02

Prerequisites: CA151(Data Structures Lab) .

Course Objectives

At the end of the course the students will understand

- To build software development skills using java programming for real world applications.
- To implement GUI applications
- To implement concept of process synchronization.
- To learn the network programming.\
- To learn server side program.

Course Outcomes

At the end of the course the students will be able to

- Develop object-oriented concepts using java programming
- Create the real-time applications.
- Develop networking applications.
- Develop server side applications using java servelets.

List of Programs

1. Write a program that displays the roots of a quadratic equation $ax^2+bx+c=0$. Calculate the discriminant D and basing on the value of D, describe the nature of roots.
2. Write a menu driven Program to perform the various Bitwise operators.
3. Write a program to perform the following operations on single dimensional array.
 - a) Find minimum and maximum elements.
 - b) Sum of the elements.
 - c) Sort given list of number.
 - d) Search for an element.
 - e) Remove duplicates.
4. Write a program to implement possible operations on matrix.
5. Write a program to implement the following on strings.
 - a) Reverse of the given string.
 - b) Check whether the given sting is Palindrome or not.
 - c) Comparison of two strings.
 - d) Sort an array of strings
 - e) Sort an array of strings using command line arguments.
6. Write a Program to define a student class, describe its constructor, overload the Constructors and instantiate its object.
7. Program to implement operations on complex numbers (Passing object to methods and use 'this' keyword).
8. Write a program to Implement the following bank operations (Use array of objects)
 - a. withdraw
 - b. deposit
 - c. Transfer
9. Write a program that displays area of different Figures (Rectangle, Square, Triangle) using the method overloading.

10. Write a Java program that counts the number of objects created by using static variable and static method.
11. Write a program that uses both recursive and non-Recursive methods to print the nth value of the Fibonacci sequence.
12. Write a program to perform the following:
 - a) Call by Value b) Call by Reference.
13. Write a program to create a player class and inherit three sub classes Cricket_Player, Hockey_Player, FootBall_Player.
14. Write a program to implement method overriding.
15. Write a program to give a simple example for abstract class.
16. Write a program to implement finals, blank finals, final methods and final class. Write observations.
17. Write a program to find the details of the students eligible to enroll for the examination (Students, Department combinedly give the eligibility criteria for the enrollment class) using interfaces.
18. Write program to implement a package.
19. Write a program that displays number of characters, lines and words in a text file.
20. Write a program to implement User defined Exception and handle that exception.
21. Write a program to create thread life cycle.
22. Write a program that displays the time continuously using threads.
23. Write a program to develop a producer and consumer problem using Thread.
24. Write an Applet program to create a simple calculator to perform addition, subtraction, multiplication and division using Button, Label and TextField components.
25. Write a program to implement client-server communication using (i) TCP (ii) UDP
26. Write a program on Generic servlets and HTTP servlet.
27. Demonstrate cookie and session management in servlets.

Web References

1. <http://www.cplusplus.com/reference/>
2. <http://en.cppreference.com/w/>
3. <http://www.decompile.com/>
4. <http://www.programmingsimplified.com/cpp>
5. <http://www.learncpp.com/>
6. <http://www.stroustrup.com/>

CA162 : WEB TECHNOLOGIES LAB

Lectures	: ---	Sessional Marks	: 40
Practicals	: 6 Periods/week	Sem.End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 02

Prerequisite: Problem Solving with C

Course Objectives

At the end of the course the students will understand

- Basic technologies to develop web documents.
- Dynamic HTML Pages and Event handling mechanism.
- XML and JDBC.
- PHP technologies.

Course Outcomes

At the end of the course the students will be able to

- Create static web pages using HTML, CSS, and JavaScript.
- Design dynamic WebPages using client side scripting.
- Create XML documents and to create web applications with JDBC.
- Write server side programs with PHP Technologies.

List of Programs

1. Develop and demonstrate a HTML5 document that illustrates the use of ordered list, unordered list, table, borders, padding, color, and the <div> & tag.
2. Write HTML5 code to provide intra and inter document linking.
3. Create a web page with the following using HTML5:
 - a. To embed an image map in a web page
 - b. To fix the hot spots
 - c. Show all the related information when the hot spots are clicked
4. Create a web page with all types of Cascading style sheets.
5. Create a web page with the following using CSS:
 - a. Text shadows, rounded corners and box shadows.
 - b. Linear and Radial gradients.
 - c. Animation
 - d. Transitions and Transformations.
6. Create a HTML5 form that interacts with the user. Collect first name, last name and date of birth and display that information back to the user.
7. Develop a HTML5 Form, which accepts any Mathematical expression. Write JavaScript code to evaluate the expression and Displays the result.
8. Create a HTML5 form that has number of Textboxes. When the form runs in the Browser fill the textboxes with data. Write JavaScript code that verifies that all textboxes has been filled. If a textboxes has been left empty, popup an alert indicating which textbox has been left empty.
9. Create a home page for “Cyber book stores” that will display the various books available, the authors and prices of the books. Include a list box that contains various subjects and a “submit” button, which displays information about the books on the subject required by the user.

10. Create a bank entry form using appropriate form elements. The account number must not be visible on the screen. The name and address must be stored in one place. There must be a text box showing the opening balance of the customer. The user should be able to make a choice of either a deposit (or) withdrawal transaction. Accordingly, when the user deposits (or) withdraws money, the opening balance must be updated using CREDIT/DEBIT button. The user should not be able to make any entries in the opening balance text box.
11. Using functions, write a JavaScript code that accepts user name and password from user. Check their correctness and display appropriate alert messages. Restrict the user to try only for a maximum of three times.
12. Modify the above program that suits the following requirements:
 - a. After a lapse of 15 seconds, the password should be generated automatically.
 - b. For impatient users, place a button that displays the passwords immediately without waiting for 15 seconds
13. Create an HTML5 file for registration with three text fields name, mobile number and address. Write JavaScript to validate name, mobile number and address. Mobile number should be of 10 digits. Show alert message when user enter invalid entity.
14. Write a JavaScript code block using arrays and generate the current date in words, this should include the day, month and year.
15. Write a program to display a form that accepts student's name, age, father name. When age field receives its focus display message that age should be 18 to 25. After losing its focus from age field verify user entered in between given values or not display respective message
16. Create a web page using two image files, which switch between one another as the mouse pointer moves over the images. Use the mouseover and mouseout event handlers.
17. Perform the following using JavaScript
 - a. To update the information into the array, in the "Click" event of the button "Update".
 - b. To sort the elements of an array (Use array object)
 - c. To find the duplicate elements of an array.
18. Demonstrate the following:
 - a. String and Math objects
 - b. Alphabetic and Numeric fields
 - c. Calendar object.
19. Write an XML file which displays the book details that includes the following:
 - 1) Title of book
 - 2) Author name
 - 3) Edition
 - 4) PriceWrite a DTD to validate the above XML file and display the details in a table (to do this use XSL).
20. Design an XML document to store information about a student in an engineering college affiliated to ANU. The information must include college id, Name of the

College, Branch, Year of Joining, and e-mail id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.

21. Create an XML document, which contains 10 users information. Implement a program, which takes User Id as an input and returns the user details by taking the user information from the XML document
22. Create tables in the database which contain the details of items (books in our case Like Book name, Price, Quantity, Amount) of each category. Modify your catalogue page in such a way that you should connect to the database and extract data from the tables and display them in the catalogue page using JDBC.
23. Write a program on simple PHP using arithmetic operators.
24. Write a program for form processing and business logic.
25. Demonstrate Cookie and Session Management in PHP.

Web References

1. www.w3schools.com
2. www.tutorialspot.com
3. www.deitel.com

CA163 : PYTHON PROGRAMMING LAB

Lectures	: ---	Sessional Marks	: 40
Practicals	: 6 Periods/week	Sem.End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 02

Prerequisite: Problem Solving with C

Course Objectives

The objectives of the course are to:

- Introduce the C++ fundamentals and Object oriented features using C++
- Introduce fundamentals of Python Programming language.
- Teach students Lists, defining functions in Python.
- Demonstrate object orientation concepts , mutable immutable types in Python

Course Outcomes

At the end of the course the students will be able to

- Solve the problems using C++
- Explain the fundamentals of Python programming language.
- Solve the problems in Python by defining modules,files, exception handling.
- Compare and contrast mutable and immutable datatypes and can demonstrate object orientation in Python in solving real world problems

CYCLE - I

1. Create a class RATIONAL and perform various operations on two Rational numbers using operator overloading.
2. Simulate Credit card transactions (use Inheritance).
3. Demonstrate Polymorphism using Shape objects in c++

CYCLE - II

4. Print all Prime numbers in an Interval
5. Program for array rotation
6. Program to Split the array and add the first part to the end
7. program to find second largest number in a list
8. program to multiply two matrices

CYCLE - III

9. Check if a Substring is Present in a Given String
10. Replace duplicate Occurrence in String
11. Print anagrams together in Python using List and Dictionary
12. Counting the frequencies in a list using dictionary in Python
13. Join Tuples if similar initial element
14. Extract digits from Tuple list
15. Get number of characters, words, spaces and lines in a file
16. Program for Sieve of Eratosthenes

Web References

1. <https://www.w3schools.com/cpp/>
2. <https://www.w3schools.com/python/>
3. <https://www.tutorialspoint.com/cplusplus/index.htm>
4. <https://www.tutorialspoint.com/python/index.htm>
5. https://onlinecourses.nptel.ac.in/noc20_cs70/preview

II Year MCA – I Semester**CA211 : DATA MINING AND BIG DATA**

Lectures +Tutorial	: 3+1Periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 04

Prerequisite: Basic Knowledge in Database Management Systems and Object Oriented Programming Languages.

Course Objectives

- To understand the Data Mining techniques and general data features.
- To learn data preprocessing and association rule mining techniques.
- To know about classification & Clustering techniques..
- To understand the need and application of Map Reduce and various search algorithms applicable to Big Data

Course Outcomes

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

- Explain the fundamental concepts of data mining and features of data.
- Discover and measure interesting patterns from different kinds of databases.
- Demonstrate different classification and clustering techniques.
- Design algorithms by employing Map Reduce technique for solving Big Data problems.

UNIT- I**15 Periods**

Introduction: Why Data Mining - What Is Data Mining? - What Kinds of Data Can Be Mined?- What Kinds of Patterns Can Be Mined? - Which Technologies Are Used? - Which Kinds of Applications Are Targeted? - Major Issues in Data Mining

Getting to Know Your Data: Data Objects and Attribute Types, Basic Statistical Descriptions of Data, Data Visualization, Measuring Data Similarity and Dissimilarity

UNIT- II**15 Periods**

Data Preprocessing: An Overview, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization.

Mining Frequent Patterns, Associations, and Correlations - Basic Concepts and Methods: Basic Concepts, Frequent Itemset Mining Methods, Which Patterns Are Interesting?—Pattern Evaluation Method.

UNIT- III**15 Periods**

Classification - Basic Concepts: Basic Concepts, Decision Tree Induction, Bayes Classification Methods, Rule-Based Classification, Model Evaluation and Selection, Techniques to Improve Classification Accuracy.

Cluster Analysis - Basic Concepts and Methods: Cluster Analysis, Partitioning Methods, Hierarchical Methods, Density-Based Methods, Grid-Based Methods, Evaluation of Clustering.

Outlier Detection: Outliers and Outlier Analysis, Outlier Detection Methods.

UNIT- IV**15 Periods**

Big Data: Introduction – distributed file system – Big Data and its importance, Four Vs, Drivers for Big data, Big data analytics, Big data applications.

Hadoop: Hadoop Architecture, Hadoop Storage: HDFS, Common Hadoop Shell commands, Anatomy of File Write and Read, Hadoop MapReduce paradigm, Writing Hadoop MapReduce Programs.

Prescribed Books :

1. Jiawei Han MichelineKamber, —Data mining – Concepts and Techniques, Morgan Kaufmann publishers, Third Edition, 2012.
2. Boris Iubinsky, Kevin t. Smith, Alexey Yakubovich, —Professional Hadoop Solutions, Wiley, ISBN: 9788126551071, 2015.
3. Chris Eaton, Dirk deroos et al., —Understanding Big data, McGraw Hill, 2012.
4. Tom White, —HADOOP: The definitive Guide, O Reilly 2012.

Reference Book

1. Frank J.Ohlhorst, —Big Data Analytics: Turning Big Data Into Big Money,2nd Edition, TMH,2012.
2. Seema Acharya, Big Data and Analytics, Wiley, 2015.

Web References

1. http://www.tutorialspoint.com/data_mining/
2. https://www.tutorialspoint.com/big_data_tutorials.htm
3. <http://www.datasciencecentral.com/profiles/blogs/17-short-tutorials-all-datascientists-should-read-and-practice>
4. <http://nptel.ac.in/courses/110106064/>

CA212 : ARTIFICIAL INTELLIGENCE

Lectures	: 4 Periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 04

Prerequisite: CA111(Data Structures), Problem solving with C.

Course Objectives

- To develop the student's understanding of the issues involved in trying to define and simulate intelligence.
- To familiarize the student with specific, well known Artificial Intelligence methods, algorithms and results.
- To provide the student additional experience in the analysis and evaluation of complicated systems.
- To provide the student with paper and proposal writing experience.

Course Outcomes

- Understand the fundamental concepts of artificial intelligence and ability to apply problem solving techniques for solving simple AI problems.
- Ability to represent the given natural language sentences in predicate/proposition logic and infer new knowledge using forward/ backward reasoning.
- Ability to represent the given natural language information as weak or strong slot-and-filler structures.
- Ability to get familiarized to various planning techniques and understand the concepts of connectionist models and expert systems.

UNIT- I**15 Periods****What is AI?**

The AI Problems, The Underlying Assumption, What is AI Technique?, The level of the Model, Criteria for Success.

Problems, Problem spaces & Search

Defining the Problem as a State Space Search, Production Systems, Problem Characteristics, Production System Characteristics, Issues in the design of Search Programs, Additional Problems.

Heuristic search techniques

Generate and Test, Hill Climbing, Best First Search, Problem Reduction, Constraint Satisfaction, Means Ends Analysis.

UNIT- II**15 Periods****Knowledge Representation Issues**

Representations and Mappings, Approaches to Knowledge Representation, Issues in Knowledge Representation, The Frame Problem.

Using Predicate Logic

Representing Simple Facts in Logic, Representing Instance and Isa Relationships, Computable Functions and Predicates, Resolution, Natural Deduction.

Representing knowledge using Rules

Procedural versus Declarative Knowledge, Logic Programming, Forward versus Backward Reasoning, Matching, Control Knowledge

UNIT- III

15 Periods

Weak slot & filler Structures, Semantic Nets, Frames

Planning

Overview, An Example Domain : The Blocks World, Components of a Planning System, Goal Stack Planning, Nonlinear Planning Using Constraint Posting, Hierarchical Planning, Reactive Systems, Other Planning Techniques .

UNIT- IV

15 Periods

Natural Language Processing

Introduction, Syntactic Processing, Semantic Analysis, Discourse and Pragmatic Processing.

Expert Systems

Representing and Using Domain Knowledge, Expert System Shells, Explanation, Knowledge Acquisition

Prescribed Book

1. Knight K, "Artificial Intelligence", TMH (Third Edition)

Reference Books

1. Michael Negnevitsky, "Artificial Intelligence – A Guide to Intelligent Systems", Third Edition, Pearson Education.
2. Winston P.H, "Artificial Intelligence", Addison Wesley (1993)

Web References

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

CA213 : CLOUD COMPUTING

Lectures	: 4 Periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 04

Prerequisite: CA114(Computer Networks), CA122 (Web Technologies)

Course Objectives

- To understand the concepts of Cloud Computing.
- To learn Taxonomy of Virtualization Techniques and acquire knowledge on Aneka Cloud Application Platform
- To understand concurrent, task and data-intensive computing.
- To learn Industry Cloud Platforms.

Course Outcomes

At the end of the course student will:

- Understand the key dimensions of the challenge of Cloud Computing.
- Recognize the concept of virtualization, how this has enabled the development of Cloud Computing and use of Cloud Application Platform.
- Know parallelism, thread, task, map-reduce programming.
- Explore some important cloud applications and cloud computing driven commercial systems.

UNIT- I**12 Periods**

Introduction: Cloud computing at a glance, Historical Developments, Building Cloud Computing Environments, Computing Platforms and Technologies.

Principles of Parallel and Distributed Computing: Eras of Computing, Parallel Vs Distributed computing, Elements of Parallel Computing, Elements of Distributed Computing, Technologies for Distributed Computing.

UNIT- II**18 Periods**

Virtualization: Introduction, Characteristics of Virtualized Environments, Taxonomy of Virtualization Techniques, Virtualization and Cloud Computing, Pros and Cons of Virtualization, Technology Examples.

Cloud Computing Architecture: Introduction, Cloud reference model, Types of clouds, Economics of the cloud, open challenges.

Aneka: Cloud Application Platform: Framework Overview, Anatomy of the Aneka Container, Building Aneka Clouds, Cloud programming and Management.

UNIT- III**18 Periods**

Concurrent Computing: Thread Programming: Introducing Parallelism for Single machine Computation, Programming Application with Threads, Multithreading with Aneka, Programming Applications with Aneka Threads.

High-Throughput Computing: Task Programming: Task Computing, Task-based Application Models, Aneka Task-Based Programming.

Data Intensive Computing: Map-Reduce Programming: What is Data-Intensive Computing, Technologies for Data-Intensive Computing, Aneka MapReduce Programming.

UNIT-IV**12 Periods**

Cloud Platforms in Industry: Amazon Web Services, Google AppEngine, Microsoft Azure, Observations.

Cloud Applications: Scientific Applications, Business and Consumer Applications.

Advanced Topics in Cloud Computing: Energy Efficiency in Clouds, Market Based Management of Clouds , Federated Clouds/ InterCloud, Third Party Cloud Services.

Prescribed Book

1. Raj kumarBuyya, Christian Vecchiola, S.ThamaraiSelvi, "Mastering Cloud Computing", McGraw Hill Education (2013).

Reference Books

1. Raj kumarBuyya, James Broberg, AndrzejGoscinski, "Cloud Computing Principles and Paradigms", Wiley Publications.
2. Michael Miller, "Cloud Computing", Pearson Education.
3. Haley Beard, "Cloud Computing Best Practices for Managing and Measuring Processes for On-demand Computing, Applications and Data Centers in the Cloud with SLAs", Emereo Pty Limited, July 2008.
4. George Reese, "Cloud Application Architectures", ISBN: 8184047142,Shroff/O'Reilly, 2009.
5. KailashJayaswal, JagannathKallakurchi, Donald J. Houde, Dr. Deven Shah, "Cloud Computing Black Book", Dreamtech Press.

Web References

1. <https://www.webopedia.com/cloud/>
2. <https://technet.microsoft.com/en-us/magazine/hh509051.aspx>

CA214(A) : INTERNET OF THINGS

Lectures +Tutorial	: 3+1Periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 04

Prerequisite: Embedded Systems.

Course Objectives

- Classify the interconnection and integration of the physical world and IoT devices
- Interpret the various IoT applications and its infrastructures
- Relate the concept of setting up IOT Devices with Python.
- Describe the concepts of interfacing hardware to develop IoT projects.

Course Outcomes

- Understand building blocks of Internet of Things, characteristics and the application areas of IOT.
- Learn Programming techniques for IOT Devices and Have real time experience with Raspberry Pi kit to interface various devices.
- Remotely monitor data and control devices.
- Develop real life IoT based projects.

UNIT – I**15 Periods**

Introduction to Internet of Things: introduction, Physical Design of IoT, Logical Design of IoT, IoT Enabling Technologies, IoT Levels & Deployment Templates.

Domain Specific IoTs: Introduction, Home Automation, Cities, Environment, Energy, Retail, Logistics, Agriculture, Industry, Health & Lifestyle.

UNIT – II**15 Periods**

IoT and M2M: Introduction, M2M, Difference between IoT and M2M, SDN and NFV for IoT.

IoT System Management with NETCONF – YANG: Need for IoT Systems Management, SNMP, Network Operation Requirements, NETCONF, YANG, IoT Systems Management with NETCONF-YANG.

UNIT – III**15 Periods**

IoT Platforms Design Methodology: Introduction, IoT Design Methodology, Case Study on IoT System for Weather Monitoring, Motivation for Using Python, Python Packages of interest for IoT.

IoT physical Devices & Endpoints: What is an IoT Device, Exemplary Device- Raspberry Pi, About the Board, Linux on Raspberry Pi, Raspberry Pi interfaces, Programming Raspberry Pi with Python.

UNIT – IV**15 Periods**

IoT Physical Servers & Cloud Offerings: Introduction of Cloud Storage Models & Communication APIs, WAMP – AutoBahn for IoT, Xively Cloud for IoT, Python Web Application Framework – Django, Designing a RESTful Web API, Amazon Web Services for IoT, SkyNetIoT Messaging Platform

Case Studies Illustrating IoT Design: Introduction, Home Automation, Cities, Environment, Agriculture, Productivity Applications

Prescribed Book

1. ArshdeepBahga, Vijay Madiseti: Internet of Things - A Hands on Approach, Universities Press, 2015.

Reference Books

1. Dr. SRN Reddy, RachitThukral and Manasi Mishra, "Introduction to Internet of Things: A practical Approach", ETI Labs
2. Pethuru Raj and Anupama C. Raman, "The Internet of Things: Enabling Technologies, Platforms, and Use Cases", CRC Press
3. Jeeva Jose, "Internet of Things", Khanna Publishing House, Delhi
4. Adrian McEwen, "Designing the Internet of Things", Wiley
5. Raj Kamal, "Internet of Things: Architecture and Design", McGraw Hill
6. CunoPfister, "Getting Started with the Internet of Things", O Reilly Media

Web References

1. [https://en.wikipedia.org/wiki/Internet_of Things](https://en.wikipedia.org/wiki/Internet_of_Things)
2. www.iot-a.eu/

CA214(B) : MACHINE LEARNING

Lectures +Tutorial	: 3+1Periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 04

Prerequisites: CA123 (Object Oriented Programming).

Course Objectives

- To understand how machine learning is used to solve problems in real world and the concept of descriptive analytics.
- Understand the concept of simple and multiple linear regression and its applications in predictive analytics.
- Understand the foundations of machine learning algorithms.
- Understand recommender systems and their business applications.

Course Outcomes

On completion of the course, students will be able to

- Understand types of machine learning algorithms and framework for building machine learning models and understand the concept of descriptive analytics.
- Implement simple linear regression and multiple linear regression, logistic regression algorithms and various classification algorithms.
- Apply machine learning algorithms available in Scikit-learn (sklearn) to the real world problems.
- Build and evaluate recommendation systems using Python libraries and develop models for sentiment classification.

UNIT- I**15 Periods**

Introduction To Machine Learning: Introduction to Analytics and Machine Learning, Why Machine Learning? , Framework for Developing Machine Learning Models
Descriptive Analytics: Working with Data Frames in Python, Handling Missing Values, Exploration of Data using Visualization.

UNIT-II**15 Periods**

Linear Regression: Simple Linear Regression, Steps in Building a Regression Model, Building Simple Linear Regression Model, Multiple Linear Regression Classification Problems: Classification Overview, Binary Logistic Regression, Credit Classification, Gain Chart and Lift Chart.

UNIT-III**15 periods**

Advanced Machine Learning: Overview, Gradient Descent Algorithm, Scikit-Learn Library for Machine Learning, Advanced Machine Learning Algorithms
Forecasting: Forecasting overview, Components of Time-Series Data, Moving Average, Decomposing Time-Series, Auto-Regressive Integrated Moving Average Models.

UNIT- IV**15 Periods**

Recommender Systems: Overview, Collaborative Filtering, Using Surprise Library, Matrix Factorization

Text Analytics: Overview, Sentiment Classification, Naïve-Bayes Model for Sentiment Classification, Using TF-IDF vectorizer , Challenges of Text Analytics

Prescribed Book

1. ManaranjanPradhan , U Dinesh Kumar, Machine Learning using Python, Wiley

Reference Books

1. Andreas C. Müller, Sarah Guido, Introduction to Machine Learning with Python: A Guide for Data Scientists, O 'Reilly Media, Inc.,2016
2. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and Tensor Flow: Concepts, Tools, and Techniques to Build Intelligent Systems, O'Reilly Media, Inc,2019
3. Mark E. Fenner, Machine Learning with Python for Everyone, Addison-Wesley Data & Analytics Series.
4. Python Machine Learning; Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow, Packt publishing.

Web References

1. https://www.w3schools.com/python/python_ml_getting_started.asp
2. <https://www.geeksforgeeks.org/introduction-machine-learning-using-python/>
3. https://www.tutorialspoint.com/machine_learning_with_python/index.htm
4. <https://pythonprogramming.net/machine-learning-tutorials/>
5. <https://machinelearningmastery.com/machine-learning-in-python-step-by-step/>

CA214(C) : MOBILE APPLICATION DESIGN AND DEVELOPMENT

Lectures +Tutorial	: 3+1Periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 04

Prerequisites: Basic Programming

Course Objectives

At the end of the course the students will understand

- Characteristics of mobile applications.
- Intricacies of UI required by mobile applications.
- design aspects of mobile application.
- development and programming of mobile applications.

Course Outcomes

At the end of the course the students will be able to

- Understand the Android application architecture, including the roles of the task stack, activities, and services.
- Build user interfaces with fragments, views, form widgets, text input, lists, tables, and more.
- Use advanced UI widgets for scrolling, tabbing, and layout control and Present menus via the Android action bar.
- Handle menu selections and to develop advanced mobile applications that accesses the databases and the web

UNIT – I**15 Periods**

GETTING STARTED WITH ANDROID: Android introduction, Versions of android, Features of android, Architecture, Devices in the market, Developer community.

ACTIVITIES, FRAGMENTS, INTENTS: Understanding activities, Applying Styles and Themesto an Activity, Hiding the Activity Title.

UNIT – II**15 Periods**

ACTIVITIES, FRAGMENTS, INTENTS: Displaying a Dialog Window, Displaying a Progress Dialog, Linking activities using intents, Fragments, Fragments-Adding Fragments Dynamically, Life Cycle of a Fragment, Interactions between Fragments, Calling built in apps using intents, Understanding the Intent Object, Using Intent Filters, Displaying Notifications.

UNIT – III**15 Periods**

GETTING TO KNOW ANDROID UI: Understanding the components of screen - Views and view groups, Liner layout, Absolute layout, Table layout, Relative layout, Frame layout, Scroll view.

DISPLAY ORIENTATION: Anchoring views, Resizing and repositioning views, Managing changes to screen orientation, Utilizing the action bar, Creating UI programmatically.

UNIT – IV**15 Periods**

DESIGNING UI WITH VIEWS: Using basic views - Text view, Button, Image Button, Edit text, check Box, Toggle button, Radio button, and Radio group views, Progress

bar view and Auto complete text view, Using List Views to Display Long Lists-ListView View, Using the Spinner View.

Prescribed Book

1. Wei-MengLee, "Beginning Android 4 Application Development", Wiley India (Wrox).

Reference Books

1. Reto Meier, "Professional Android 4 Application Development", Wiley India, (Wrox).
2. James C Sheusi, "Android Application Development for Java Programmers", Cengage Learning.
3. Sam's "Teach Yourself Android Application Development in 24 Hrs", Lauren Darcy and Shane Conder, 2/e.
4. Wallace Jackson, "Android apps for absolute beginners", Apress.

Web References

1. <http://blogs.msdn.com/b/ie/archive/2010/11/17/html5-and-real-worldsite-performance-seventh-ie9-platform-preview-available-fordevelopers.aspx>
2. https://www.tutorialspoint.com/mobile_development_tutorials.htm
3. <https://www.tutorialspoint.com/android/>
4. <https://developer.android.com/training/basics/firstapp/creatingproject.html>

CA 214(D) : IMAGE PROCESSING

Lectures +Tutorial	: 3+1 periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 04

Prerequisites: Basic Math and Programming

Course Objectives

- Study the image fundamentals and mathematical transforms necessary for image processing
- To instigate the various image filtering techniques
- To originate the various image restoration and compression techniques
- To study the image segmentation, morphology and color image representation.

Course Outcomes

Upon Completion of the course, student will be able to

- Understand the various mathematical transforms of a digital image
- Understand various image filtering methods
- Realize the image enhancement, image restoration and image compression techniques
- Bring out morphology, image segmentation and representation techniques
- Solve problems of Image processing

UNIT – I**12 Periods**

What Is Digital Image Processing?, Background on MATLAB and the Image Processing Toolbox, Fundamentals- The MATLAB Desktop, Using the MATLAB Editor/Debugger, Saving/Retrieving Work session data, Digital Image Representation, Image I/O and display, Classes and image types, M-function programming

UNIT – II**15 Periods**

Intensity Transformations and Spatial Filtering: Intensity Transformation Functions, Histogram Processing and function plotting, Spatial Filtering, Image processing toolbox standard spatial filters.

Filtering in the Frequency Domain: The 2-D Discrete Fourier Transform, Computing And Visualizing 2-D Discrete Fourier Transform, Filtering in the Frequency Domain, Obtaining Frequency Domain Filters from spatial filters, Generating Filters Directly In The Frequency Domain, Highpass Frequency Domain Filters.

UNIT – III**16 Periods**

Image Restoration: A Model of the Image Degradation/Restoration Process, Noise models, Restoration in the presence of noise only- Spatial Filtering, Periodic Noise Reduction by Frequency Domain filtering. Modeling the Degradation function, Direct inverse filtering, Wiener filtering, Constrained Least Squares Filtering. **Image Compression:** Background, Coding Redundancy, Spatial Redundancy, Irrelevant Information, JPEG and Video compression.

UNIT – IV**15 Periods**

Morphological Image Processing: Preliminaries, Erosion and Dilation, Combining Erosion and Dilation, Labeling connected components, Morphological Reconstruction,

Gray-Scale morphology. **Image Segmentation:** Point, Line and Edge Detection, Line Detection using Hough Transform, Thresholding, Region-Based Segmentation
Color Image Processing: Color Image representation

Prescribed Book

1. Rafael C. Gonzalez, Richard E. Woods, 'Digital Image Processing' Pearson Education, 3rd Edition.

Reference Books

1. Rafael C Gonzalez, Richard E. Woods, Steven L. Eddins, 'Digital Image Processing Using Matlab', 2nd Edition.
2. Milan Sonka, Vaclav Hlavac, Roger Boyle, 'Image Processing, Analysis, and Machine Vision', 3rd Edition.
3. A.K.Jain, 'Fundamentals of Digital Image Processing' PHI.

Web References

1. <http://www.imageprocessingbasics.com/>
2. www.imageprocessingplace.com/root_files_V3/tutorials.htm
3. www.library.cornell.edu/preservation/tutorial/intro/intro-01.html
4. www.olympusmicro.com/primer/digitalimaging/javaindex.html
5. <http://www.mathworks.com>
6. <https://www.cs.washington.edu/research/metip/tutor/tutor.html>
7. <https://www.engineersgarage.com/articles/image-processing-tutorial-applications>
8. http://www.bogotobogo.com/Matlab/Matlab_Tutorial_Digital_Image_Processing_I.php
9. <https://www.youtube.com/watch?v=3St0byWYcEI>

CA214(E) : DATA SCIENCE

Lectures +Tutorial	: 3+1Periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 04

Prerequisites: Basic Programming and Statistics.

Course Objectives

- Explore the key differences between supervised learning and unsupervised learning
- Manipulate and analyze data using scikit-learn and pandas libraries
- Understand key concepts of regression
- Discover advanced techniques to improve the accuracy of your model
- Understand how to speed up the process of adding new features
- Simplify your machine learning workflow for production

Course Outcomes

By the end of this chapter, you will be able to

- Understand data science applications and implement regression using python.
- Evaluate the performance of multiclass models.
- Explore the key steps involved in performing exploratory data analysis
- Apply different dimensionality reduction techniques to large datasets, to fit models based on those datasets and analyze their results

UNIT – I**15 Peroids**

Introduction to data science in Python: Introduction, Application of Data Science, What Is Machine Learning? Python for Data Science, Scikit-Learn

Regression: Introduction, Simple Linear Regression, Multiple Linear Regression, Conducting Regression Analysis Using Python, The Correlation Coefficient, Multiple Regression Analysis.

UNIT – II**15 Periods**

Multiclass Classification with Random Forest: Introduction, Training a Random Forest Classifier, Evaluating the Model's Performance, Maximum Depth, Minimum Sample in Leaf, Maximum Features

How to Assess performance: Introduction, Splitting data, Assessing model performance with regression models, R^2 Score, Regression model, mean absolute error, Assessing Model Performance for Classification Models, The Confusion Matrix, Receiver Operating Characteristic Curve, Area Under the ROC Curve, Saving and Loading models

UNIT – III**15 Periods**

Analyzing a Dataset: Introduction, Exploring Your Data, Analyzing Your Dataset, Analyzing the Content of a Categorical Variable, Summarizing Numerical Variables, Visualizing Your Data, Boxplots

Data Preparation: Introduction, Handling Row Duplication, Converting Data Types, Handling Incorrect Values, Handling Missing Values

UNIT – IV**15 Periods**

Dimensionality Reduction: Introduction, Creating a High-Dimensional Dataset, Strategies for Addressing High-Dimensional Datasets, Comparing Different Dimensionality Reduction Techniques

Ensemble Learning: Introduction, Ensemble learning, Simple Methods for Ensemble Learning, Advanced methods for Ensemble Learning

Prescribed Book

1. Anthony So, Thomas V. Joseph, Robert Thas John, Andrew Worsley, Dr. Samuel Asare The Data Science Workshop: A New, Interactive Approach to Learning Data Science- Second Edition, Packt Publishing, 2020

Reference Books

1. Joel Grus, "Data Science from Scratch: First Principles with Python", O'Reilly Media
2. Jain V.K., "Data Science and Analytics", Khanna Publishing House, Delhi.
3. Jiawei Han and Jian Pei, "Data Mining Concepts and Techniques", Third Edition, Morgan Kaufmann Publishers
4. AurélienGéron, "Hands-On Machine Learning with Scikit-Learn and Tensor Flow: Concepts, Tools, and Techniques to Build Intelligent Systems", 1st Edition, O'Reilly Media.

Web References

1. https://www.tutorialspoint.com/python_data_science/index.htm
2. <https://www.datasciencecentral.com/>
3. <https://towardsdatascience.com/>
4. <https://www.dataquest.io/python-tutorials-for-data-science/>

CA215 (A): PROGRAMMING USING C#

Lectures +Tutorial	: 3+1Periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 04

Prerequisites CA123 (Objected Oriented Programming).

Course Objectives

- Comprehend the C# language and the .NET Framework.
- Demonstrate the use of Windows Forms applications with rich, highly responsive user.
- Identify the cloud web applications and Services using ASP.NET.
- Relate the use of Language Integrated Query (LINQ).

Course Outcomes

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

- Apply the fundamental concepts of C# programming.
- Implement advanced OOPS concepts in console applications.
- Develop and deploy cloud web applications and web services using ASP.NET and AZURE API.
- Develop database driven applications utilizing XML and LINQ.

Course Content:**UNIT- I****12 Periods**

Introducing C#, Writing a C# Program, Variables and Expressions. Flow Control, More About Variables, Functions.

UNIT - II**12 Periods**

Debugging and Error Handling, Introduction to Object-Oriented Programming, Defining Classes, Collections, Comparisons and Conversions.

UNIT- III**14 Periods**

Generics, Additional C# Techniques, Basic Desktop Programming. Advanced Desktop Programming. Advanced Cloud Programming

UNIT - IV**12 Periods**

Files, XML and JSON, LINQ, DATABASES

Prescribed Book

1. Karli Watson, Christian Nagel, Jacob Hammer Pedersen, Jon Reid, and Morgan Skinner, BEGINNING VISUAL C# 2015, Wiley Publishing, Inc.

Reference Books

1. Stephen C. Perry, Core C# and .NET, Pearson Education, 2006.
2. Herbert Scheldt, C#: The Complete Reference, TATA McGraw Hill Publishing.
3. Andrew Troelsen, Pro C# and the .NET Platform, A! Press.
4. Kevin Hoffman, Microsoft Visual C# 2005 Unleashed, Sams Pearson India.

CA215(B) : DEVOPS

Lectures +Tutorial	: 3+1Periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 04

Prerequisite: CA124(Software Engineering), Basics of Computing.

Course Objectives

- To understand the concepts of DevOps.
- To know DevOps architecture and Deployment.
- To identify monitoring and security methods.
- To recognize business considerations and operations.

Course Outcomes

Upon completion of this course, students should be able to:

- Understand the need of DevOps and operations.
- Recognize testing models.
- Identify monitoring and security tools.
- Know business understanding in software development.

UNIT- I**15 Periods**

What Is DevOps? – Introduction - Why DevOps? - DevOps Perspective - DevOps and Agile - Team Structure - Coordination – Barriers – DevOps life cycle.

The Cloud as a Platform - Introduction - Features of the Cloud - DevOps Consequences of the Unique Cloud Features.

Operations - Introduction - Operations Services - Service Operation Functions - Continual Service Improvement - Operations and DevOps.

UNIT- II**15 Periods**

Overall Architecture - Do DevOps Practices Require Architectural Change? - Overall Architecture Structure - Quality Discussion of Microservice - Architecture - Amazon's Rules for Teams - Microservice Adoption for Existing Systems.

Building and Testing - Introduction - Moving a System through the Deployment - Pipeline - Crosscutting Aspects - Development and Pre-commit Testing - Build and Integration Testing - UAT/Staging/Performance Testing - Production – Incidents.

Deployment - Introduction - Strategies for Managing a Deployment - Logical Consistency – Packaging - Deploying to Multiple Environments - Partial Deployment - Rollback – Tools.

UNIT- III**15 Periods**

Monitoring - Introduction - What to Monitor - How to Monitor - When to Change the Monitoring Configuration - Interpreting Monitoring Data - Challenges - Tools - Diagnosing an Anomaly from Monitoring Data—the Case of Platformer.com.

Security and Security Audits - What Is Security? - Threats - Resources to Be Protected - Security Roles and Activities - Identity Management - Access Control - Detection, Auditing, and Denial of Service - Development - Auditors - Application Design Considerations - Deployment Pipeline Design Considerations.

Other Ilties - Introduction - Repeatability - Performance - Reliability – Recoverability - Interoperability - Testability – Modifiability.

UNIT- IV

15 Periods

Business Considerations - Introduction - Business Case - Measurements and Compliance to DevOps Practices - Points of Interaction Between Dev and Ops.

Operations as a Process - Introduction - Motivation and Overview - Offline Activities - Online Activities - Error Diagnosis – Monitoring.

The Future of DevOps - Introduction - Organizational Issues - Process Issues - Technology Issues - What About Error Reporting and Repair? - Final Words.

Prescribed Book

1. Len Bass, Ingo Weber, Liming Zhu, “DevOps: A Software Architect’s Perspective”, Addison-Wesley (2015).

Reference Books

1. Jennifer Davis & Katherine Daniels, “Effective DevOps: Building a Culture of Collaboration, Affinity, and Tooling at Scale”, O’Reilly (2015).
2. Kim, Gene, Behr, Kevin, & Spafford, George, “The phoenix project: A novel about IT, DevOps, and helping your business win: IT Revolution Press”, ISBN 978-0988262591 (2013).

Web References

1. <https://aws.amazon.com/devops/resources/>
2. <https://theagileadmin.com/what-is-devops/>
3. <https://techbeacon.com/53-essential-resources-devops-practitioners>

CA215(C) : BLOCKCHAIN TECHNOLOGY

Lectures +Tutorial	: 3+1Periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 04

Course Objectives

The student should be made to

- Develop familiarity of current technologies, tools.
- Impart strong technical understanding of Block Chain technologies.
- Explore the Smart Contracts and Ethereum implementation strategies.
- Introduce the current scenario and practical application areas of Hyperledger.

Course Outcomes

After successful completion of the course, the students are able to:

- Discuss the block chain technology in decentralized paradigm.
- Explore cryptography and trading Applications along with their implementation strategies.
- Discuss the implementation of smart contract and Ethereum platform.
- Explain the importance and applications of Hyperledger.

UNIT – I**15 Periods**

Block Chain 101- Distributed Systems, The History of blockchain, Introduction to blockchain, Types of block chain, CAP theorem and blockchain, Benefits and limitations of blockchain,

Decentralization-Decentralization using blockchain, Methods of decentralization, Routes to decentralization, Blockchain and full eco system decentralization, Smart contract, Decentralized Organizations, Decentralized autonomous organizations, Decentralized autonomous corporations, Decentralized autonomous societies, Decentralized applications, Platforms for Decentralization.

UNIT – II**15 Periods****Cryptography**

and Technical Foundations - Introduction, Cryptographic primitives, Asymmetric Cryptography, Public and Private-keys, Financial -market and trading.

Bitcoin - Bitcoin, Transactions, Blockchain, Bitcoin Payments.

Alternative Coins – Theoretical foundations, Bitcoin limitations, Namecoin, Primecoin, Zcash.

UNIT – III**15 Periods**

Smart Contracts -History, Definition, Recardian Contracts.

Ethereum 101-Introduction, Ethereumblockchain, Elements of the Ethereumblockchain, Precompiled contracts, Accounts, Block, Ether, Messages, Mining, Clients and Wallets, Trading and investment, The Yellow paper, The Ethereum Network, Applications developed on Ethereum, Scalability and security issues .

UNIT – IV

15 Periods

Hyperledger- Projects, Hyperledger as a Protocol, Fabric, Hyperledger Fabric, Sawtooth lake, Corda.

Alternative Blockchain- Blockchains, Platforms.

Scalability and Other Challenges - Scalability, Privacy, Security.

Prescribed Book

1. Imran Bashir, “Mastering Block Chain: Distributed Ledger Technology, Decentralization and Smart Contracts Explained”, Packt Publishing.

Reference Books:

1. Melanie Swan, “Block Chain: Blueprint for a New Economy”, O’Reilly, 2015
2. Josh Thompsons, “Block Chain: The Block Chain for Beginners- Guide to Block chain Technology and Leveraging Block Chain Programming”
3. Daniel Drescher, “Block Chain Basics”, Apress; 1st edition, 2017
4. AnshulKaushik, “Block Chain and Crypto Currencies”, Khanna Publishing House, Delhi.
5. RiteshModi, “Solidity Programming Essentials: A Beginner’s Guide to Build Smart Contracts for Ethereum and Block Chain”, Packt Publishing
6. Salman Baset, Luc Desrosiers, Nitin Gaur, Petr Novotny, Anthony O’Dowd, Venkatraman Ramakrishna, “Hands-On Block Chain with Hyperledger: Building Decentralized Applications with Hyperledger Fabric and Composer”, Import, 2018

Web References

1. <https://www.euromoney.com/learning/blockchain-explained>
2. <https://www.investopedia.com/terms/b/blockchain.asp>
3. <https://builtin.com/blockchain>

CA215 (D) : SOFTWARE TESTING AND QUALITY ASSUARANCE

Lectures +Tutorial	: 3+1Periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 04

Prerequisite: CA124 (Software Engineering).

Course Objectives

- To make the students proficient in checking the software to find errors and remove them if found.
- To make the learners understand the concept that the testing is the last step before releasing software is performed at the end of software development life cycle.
- To make the learners aware of testing tools.

Course Outcomes

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

- Aware of different types of testing.
- Understand the concept of object oriented testing with examples.
- Aware of test cases preparation, management, execution and reporting using functional test tool.

UNIT – I

12 Periods

Basic Concepts and Preliminaries: Quality Revolution, Software Quality, Role of Testing, Verification and Validation, Failure, Error, Fault and Defect, Objective of Testing, Test case, Issues, Activities, Levels, Test case selection, Test Plan and Design, test tools and automation.

Unit Testing: Concept, Static, Defect Prevention, Dynamic, Mutation testing, Debugging, Tools

Control Flow Testing: Basic idea, Outline, Control flow graph, Paths in a control flow graph, path selection criteria, Generating Test input.

UNIT – II

12 Periods

Data Flow Testing: General Idea, Data Flow Anomaly, Overview of Dynamic Data Flow Testing, Data Flow Graph, Data Flow Terms, Data Flow Testing Criteria, Comparison of Data Flow Test Selection Criteria.

System Integration Test: Concept, Different Types of Interfaces and Interface Errors, System Integration Techniques, Test Plan for System Integration.

UNIT – III

12 Periods

System Test Design: Test Design Factors, Requirement Identification, Characteristics of Testable Requirements, Test Objective Identification.

System Test Execution: Basic Ideas, Modeling Defects, Preparedness to Start System Testing, Metrics for Tracking System Test, Beta Testing, System Test Report, Measuring Test Effectiveness.

Acceptance Testing: Types, Criteria, Selection, Test Plan, Test Execution, Test Report, Testing in eXtreme Programming

UNIT – IV

12 Periods

Test Team Organization: Test Groups, Software Quality Assurance Group, System Test Team hierarchy, Effective Staffing of Test Engineers, Recruiting Test Engineers.

Software Quality: Five Views of Software Quality, McCall's Quality Factors and Criteria, ISO 9126 Quality Characteristics, ISO 9000:2000 Software Quality Standard

Prescribed Book

1. Kshirasagarnaik, priyadarshitripathy, "Software Testing and Quality Assurance: Theory and Practice", Wiley.

Reference Books

1. SrinivasanDesikan, Gopalaswamy Ramesh, "Software Testing Principles and practices", Pearson Education.
2. Dr. K.V.K.K.Prasad, "Software Testing Tools", Dream Tech Press.
3. Brian Marick, "The craft of software Testing", Pearson Education.
4. Edward Kit, "Software Testing in the Real World", Pearson.
5. Meyers, "Art of Software Testing", John Wiley.

Web References

1. https://www.tutorialspoint.com/software_testing_dictionary/quality_assurance.htm
2. www.guru99.com/software-testing.html
3. www.qatutorial.com/

CA215 (E) : DESIGN AND ANALYSIS OF ALGORITHMS

Lectures +Tutorial	: 3+1 periods/week	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 04

Prerequisite: CA111(Data Structures)

Course Objectives

At the end of the course the students will understand

- The methods for designing efficient algorithms using various data structures.
- The performance of algorithms.
- The application of mathematical tools in computing to solve fundamental problems.
- The concept of classification of algorithms.

Course Outcomes

At the end of the course the students will be able to

- Compute time and space complexity of algorithms.
- Deduce the recurrence relations that describe the time complexity of recursively-defined algorithms that employ these strategies and implement real time applications based on various data structures.
- Design algorithms using divide and conquer, greedy, and dynamic programming strategies and recite algorithms that employ these strategies.
- Design algorithms using backtracking and branch and bound strategies and recite algorithms that employ these strategies and understand the P, NP, NP hard, and NP complete classes.

UNIT – I**15 Periods****Introduction**

What is Algorithm, Algorithm Specification: Pseudocode Conventions, Recursive Algorithms; Performance Analysis: Space Complexity, Time Complexity, Asymptotic notation, Performance Measurement; Randomized Algorithms: Basics of probability theory, Randomized algorithms, Identifying the repeated element, Primality Testing, Advantages and Disadvantages.

Divide - and - conquer

General Method, Binary Search, Finding Maximum and Minimum, Merge Sort, Quick sort, Selection Problem, Strassen's Matrix Multiplication,

UNIT – II**15 Periods****The Greedy Method**

The general Method, Knapsack Problem, Tree Vertex Splitting, Job sequencing with deadlines; Minimum cost spanning trees: Prim's Algorithm, Kruskal's Algorithm;, Optimal Merge patterns, Single Source shortest paths.

Dynamic Programming

The general method, Multi-stage graphs, All pairs shortest paths, Single source shortest paths, Optimal Binary Search Trees, String editing, 0/1 Knapsack, Reliability design, The traveling sales person problem, Flow shop Scheduling.

UNIT – III

15 Periods

Basic Traversal and Search Techniques

Techniques for Binary Trees, Techniques for graphs: Breadth First Search and Traversal, Depth First Search; Connected Components and Spanning Trees, Bi-connected components and DFS.

BackTracking

The general method, The 8-queens problem, sum of subsets, Graph coloring, Hamiltonian Cycles, Knapsack Problem.

UNIT – IV

15 Periods

Branch and Bound

The Method: Least Cost search, The 15 puzzle, control abstractions for LC search, Bounding, FIFO Branch - and - Bound, LC Branch and Bound; 0/1 knapsack problem: LC Branch and Bound solution, FIFO Branch and Bound solution; Traveling Sales person.

NP-Hard and NP – complex problems

Basic concepts : Non deterministic algorithms , The classes NP hard and NP complex Cooks theorem.

Prescribed Book

1. Ellis Horowitz, SartajSahni, SanguthevarRajasekaran, “Fundamentals of Computer Algorithms”, Second Edition, Universities Press

Reference Books

1. Hopcraft.J.E, Ullman.J.D, “The design and analysis of algorithms ANOVA”, First edition, Pearson publishers.
2. I.Chandra Mohan, ”Design and Analysis of Algorithms”, PHI.
3. Thomos H Cormen, Charles ELeisevson, Ronald,Revart Clifford stein, “Introduction to algorithms”, Third edition, PHI.

Web References

1. <http://nptel.ac.in/courses/106101060/>
2. <http://www.personal.kent.edu/~rmuhamma/Algorithms/algorithm.html>
3. <http://www.facweb.iitkgp.ernet.in/~sourav/daa.html>
4. https://www.tutorialspoint.com/design_and_analysis_of_algorithms/index.htm

CA251 : DATA MINING AND BIG DATA LAB

Lectures	:---	Sessional Marks	: 40
Practicals	: 6 Periods/week	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 02

Prerequisites: Problem solving with C, Data Structures.

Course Objectives

- To learn the algorithms used for various types of Data Mining Problems.
- To discover interesting patterns, analyze supervised and unsupervised models and estimate the accuracy of the algorithms.
- Exercise the data mining techniques with varied input values for different parameters.
- To understand setting up of Hadoop Cluster
- To solve problems using Map Reduce Technique
- To solve Big Data problem

Course Outcomes

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

- Ability to understand the various kinds of tools.
- Demonstrate the classification, clustering and etc. in large data sets.
- Ability to add mining algorithms as a component to the exiting tools.
- Ability to apply mining techniques for realistic data.
- Set up multi-node Hadoop Clusters
- Apply Map Reduce algorithms for various algorithms
- Design new algorithms that uses Map Reduce to apply on Unstructured and structured data

LAB CYCLE 1(Data Mining) (Using Java, WEKA or any open source data mining tool)

1. Write a program to Generate Association rules by using A priori algorithm
2. Write a program to implement naïve Bayesian classification
3. Write a program to implement k-means clustering algorithm
4. Write a program to implement k-medoids clustering algorithm
5. Write a program to implement dbscan algorithm

LAB CYCLE 2 (Hadoop)

1. Implement the following Data structures in Java a)Linked Lists b) Stacks c) Queues d) Set e) Map
2. Study and configure hadoop for big data. Use web based tools to monitor your Hadoop setup.
3. Implement the following file management tasks in Hadoop:
 - Adding files and directories
 - Retrieving files
 - Deleting files

Hint: A typical Hadoop workflow creates data files (such as log files) elsewhere and copies them into HDFS using one of the above command line utilities.

1. Run a basic Word Count Map Reduce program to understand MapReduce Paradigm.

2. Implement Matrix Multiplication with Hadoop Map Reduce

Web References

1. nptel.ac.in/courses/106106093/35
2. <https://www.cse.iitb.ac.in/infolab/Data/Talks/krithi-talk-impact.pp>
3. <https://hortonworks.com/hadoop-tutorial/hello-world-an-introduction-to-hadoop-hcatalog-hive-and-pig/https://developer.ibm.com/hadoop/docs/getting-started/tutorials/overview-tutorial/overview-lab-1-getting-started-hadoop-biginights-2/>

CA252 : INTERNET OF THINGS LAB(Elective Lab)

Lectures	: ---	Sessional Marks	: 40
Practicals	: 6 Periods/week	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 02

Course Objectives:

- Impart necessary and practical knowledge of components of Internet of Things.
- Familiarization with Arduino/Raspberry Pi and perform necessary software installation.
- Relate the concept of setting up IOT Devices with Python programming.
- Develop skills required to build real-life IoT based projects.

Course Outcomes:

- Configure and set up the Arduino/Raspberry Pi board for a given application.
- Interface I/O devices, sensors & communication modules.
- Understand enabling technologies Embedded Devices and communication protocols for Hands on activities.
- Building IoT applications using Cloud.

List of Experiments:

1. Write a program to turn ON LED for 1 sec after every 2 seconds.
2. Write a program to turn ON LED when push button is pressed or at sensor detection.
3. Write a program to print temperature and humidity readings.
4. Write a program to turn ON motor when push button is pressed.
5. Write a program to print temperature and humidity readings on it.
6. Write a program to send sensor data to Smartphone using Bluetooth.
7. Write a program to turn LED ON/OFF when '1'/'0' is received from Smartphone using Bluetooth.
8. Write a program on Arduino/Raspberry Pi to upload temperature and humidity data to thingspeak cloud.
9. Write a program on Arduino/Raspberry Pi to retrieve temperature and humidity data from thingspeak cloud.
10. To install MySQL database on Raspberry Pi and perform basic SQL queries.
11. Write a program on Arduino/Raspberry Pi to publish temperature data to MQTT broker.
12. Write a program on Arduino/Raspberry Pi to subscribe to MQTT broker for temperature data and print it.
13. Write a program to create TCP server on Arduino/Raspberry Pi and respond with humidity data to TCP client when requested.
14. Write a program to create UDP server on Arduino/Raspberry Pi and respond with humidity data to UDP client when requested.

Web References

1. www.raspberrypi.org
2. <http://forefront.io/a/beginners-guide-to-arduino/>
3. <https://www.arduino.cc/en/Tutorial/HomePage>

CA252 : MACHINE LEARNING LAB (Elective Lab)

Lectures	:---	Sessional Marks	: 40
Practicals	: 6 Periods/week	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 02

Course objectives

- Demonstrate the basic concepts and techniques of Machine Learning.
- Develop skills of using recent machine learning software for solving practical problems.
- Provide experience of doing independent study and research.

Course Outcomes:

The students should be able to:

- Understand the implementation procedures for the machine learning algorithms.
- Design Java/Python programs for various Learning algorithms.
- Apply appropriate data sets to the Machine Learning algorithms.
- Identify and apply Machine Learning algorithms to solve real world problems.

Lab Experiments:

- 1) Load IPL dataset, and compute measures of central tendency (mean, mode, median).
- 2) Using same dataset, demonstrate exploration of data using visualization.
- 3) Implement linear regression using python.
- 4) Implement multiple linear regression using python.
- 5) Use SAheart.data to implement the following:
 - a) Build a logistic regression model to predict the probability of getting *chd* (Coronary Heart Disease), that is, $P(chd = 1)$. Use all other columns except *chd* as features to build the model.
 - b) Find out which features are statistically significant in the logistic regression model built in Previous Question. Build a new logistic regression model using only the significant features.
 - c) From the logistic regression model parameters, determine which parameters affect the probability of *chd* positively and negatively.
 - d) Calculate Youden's index for all possible cut-off probabilities ranging from 0.1 to 0.5 with an interval of 0.01. Find the optimal cut-off where Youden's index is maximum.
 - e) Build a confusion matrix based on the cut-off probability found in Question 4 and report the precision and recall of the model for *chd* cases (i.e., $chd = 1$).
- 6) Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem.
- 7) Write a program to implement random forest model.
- 8) Write a program to implement movie recommender system.
- 9) Write a program to implement Naïve-Bayes model for sentiment classification.
- 10) Implement sentiment classification using python.

Reference Books

1. Andreas C. Müller, Sarah Guido, Introduction to Machine Learning with Python: A Guide for Data Scientists, O'Reilly Media, Inc., 2016

2. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and Tensor Flow: Concepts, Tools, and Techniques to Build Intelligent Systems, O'Reilly Media, Inc, 2019
3. Mark E. Fenner, Machine Learning with Python for Everyone, Addison-Wesley Data & Analytics Series.
4. Python Machine Learning; Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow, Packt

Web references

1. https://www.python-course.eu/machine_learning.php
2. https://www.tutorialspoint.com/machine_learning_with_python/machine_learning_with_python_tutorial.pdf
3. <https://pythonprogramming.net/machine-learning-tutorial-python-introduction/>

CA252 : MOBILE APPLICATION DESIGN DEVELOPMENT LAB(Elective Lab)

Lectures	:---	Sessional Marks	: 40
Practicals	: 6 Periods/week	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 02

Prerequisite: Problem solving with C, Unix and Shell Programming.

Course Objectives

- To make the learners install and set up the android application development environment.
- To make the students design apps by displaying windows and also working with intents.
- To make the students develop applications by using menus with views.
- To make the students work with data files and SQLite database.

Course Outcomes

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

- Build custom android applications.
- Design and develop apps with good user interface by using menus and views.
- Design and develop apps with pictures and action bars.
- Develop apps that store data in files and databases.

LIST OF PROGRAMS

1. Create an Android application that shows "Welcome to Android" and run it on the emulator.
2. Create an application that displays a dialog window using an activity.
3. Create an application that displays the progress of an operation.
4. Create an application that obtains result from an activity.
5. Create an application that dynamically adds a fragment.
6. Create a screen that has input boxes for Name, Address, Gender (radio buttons for male and female), Age (numeric), Date of Birth (Date Picket), State (Spinner) and a Submit button. On clicking the submit button, print all the data below the Submit Button.
7. Use (a) Linear Layout (b) Relative Layout and (c) Grid Layout or Table Layout
8. Create an application that uses the basic views of Android.
9. Develop an application that uses a menu with 3 options for dialing a number, opening a website and to send an SMS. On selecting an option, the appropriate action should be invoked using intents.

10. Create a user registration application that stores the user details in a database table.
11. Create a database and a user table where the details of login names and passwords are stored. Insert some names and passwords initially. Now the login details entered by the user should be verified with the database and an appropriate dialog should be shown to the user.

Reference Books

1. Beginning Android 4 Application Development, Wei-Meng Lee, Wiley India (Wrox)
2. Professional Android 4 Application Development, Reto Meier, Wiley India, (Wrox)
3. Android Application Development for Java Programmers, James C Sheusi, Cengage Learning,
4. Sams Teach Yourself Android Application Development in 24 Hrs, Lauren Darcy and Shane Conder, 2nd ed.
5. Android apps for absolute beginners, Wallace Jackson, Apress.
6. Introduction to Android application development, Joseph Annuzzi, Lauren Darcey, Shane Conder, 4ed.
7. Android apps with Eclipse, OnurCinar.

Web References

1. <http://blogs.msdn.com/b/ie/archive/2010/11/17/html5-and-real-worldsite-performance-seventh-ie9-platform-preview-available-fordevelopers.aspx>
2. https://www.tutorialspoint.com/mobile_development_tutorials.htm
3. <https://www.tutorialspoint.com/android/>
4. <https://developer.android.com/training/basics/firstapp/creatingproject.html>

CA252 : IMAGE PROCESSING(Elective Lab)

Lectures	: --	Sessional Marks	: 40
Practicals	: 6 periods/week	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 02

Prerequisites: Basic Math and Programming

Course Objectives

- Perform enhancing operations on the image using spatial filters and frequency domain filters.
- Use transforms and analyse the characteristics of the image.
- Perform segmentation operations in the images.
- Estimate the efficiency of the compression technique on the images.
- Apply image processing technique to solve real health care problems.

Course Outcomes

At the end of the course, the student should be able to:

- Perform enhancing operations on the image using spatial filters and frequency domain filters.
- Use transforms and analyse the characteristics of the image.
- Perform segmentation operations in the images.
- Estimate the efficiency of the compression technique on the images.
- Apply image processing technique to solve real health care problems.

CYCLE 1

1. Image sampling and quantization
2. Analysis of spatial and intensity resolution of images.
3. Intensity transformation of images.
4. DFT analysis of images
5. Transforms (Walsh, Hadamard, DCT, Haar)
6. Histogram Processing and Basic Thresholding functions
7. Image Enhancement-Spatial filtering
8. Image Enhancement- Filtering in frequency domain

CYCLE 2

9. Image segmentation – Edge detection, line detection and point detection.
10. Basic Morphological operations.
11. Region based Segmentation
12. Segmentation using watershed transformation
13. Analysis of images with different color models.
14. Image compression techniques
15. Image restoration

CYCLE 3

16. A mini project based on medical image processing

Reference Books

1. Rafael C Gonzalez, Richard E. Woods, Steven L. Eddins, 'Digital Image Processing Using Matlab', 2nd Edition.
2. Milan Sonka, Vaclav Hlavac, Roger Boyle, 'Image Processing, Analysis, and Machine Vision', 3rd Edition.
3. A.K.Jain, 'Fundamentals of Digital Image Processing' PHI.

Web References

1. <http://www.imageprocessingbasics.com/>
2. www.imageprocessingplace.com/root_files_V3/tutorials.htm
3. www.library.cornell.edu/preservation/tutorial/intro/intro-01.html
4. www.olympusmicro.com/primer/digitalimaging/javaindex.html

CA252 : DATA SCIENCE LAB(Elective Lab)

Lectures	:---	Sessional Marks	: 40
Practicals	: 6 Periods/week	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: 3 hours	Credits	: 02

Prerequisites: CA123, CA163

Course Objectives

- To introduce core programming basics required for data science using Python language
- To introduce regression analysis and model building.
- To build skills to develop self-learning algorithms using training data to classify or predict the outcome of future datasets.
- To advance practical experience in programming tools for data sciences.
- To empower students with tools and techniques used in data science

Course Outcomes

Upon completion of course, students should be able to

- Choose and apply tools and methodologies to solve data science tasks in real world contexts and will communicate these solutions effectively.
- Interpret data, extract meaningful information, and assess findings.
- Help non-technical professionals visualize, explore, and act on data science findings.
- Integrate data science capabilities into the formation of a situation analysis.

Lab Programs

1. Write a programme in Python to predict the class of the flower based on available attributes.
2. Write a programme in Python to predict if a loan will get approved or not.
3. Write a programme in Python to predict the traffic on a new mode of transport.
4. Write a programme in Python to predict the class of user.
5. Write a programme in Python to indentify the tweets which are hate tweets and which are not.
6. Write a programme in Python to predict the age of the actors.
7. Mini project to predict the time taken to solve a problem given the current status of the user.

Reference Books:

1. Nelli, F., Python Data Analytics: With Pandas, NumPy, and Matplotlib, Second Ed., Apress, 2018.
2. Samir Madhavan, Mastering Python for Data Science, Packt Publishing Ltd., 2015.

Web References

1. https://www.tutorialspoint.com/python_data_science/index.htm
2. <https://www.datasciencecentral.com/>
3. <https://towardsdatascience.com/>
4. <https://www.dataquest.io/python-tutorials-for-data-science/>

CA253 : MINI PROJECT WORK

Lectures	: ---	Sessional Marks	: 100
Practicals	: 4 Periods/week	Sem. End Exam Marks	: ---
Sem. End Exam Duration	: ---	Credits	: 02

Prerequisites: Basic knowledge in software development phases.

Course Objectives

- To transform the theoretical knowledge into application software.
- To gain experience in organization and implementation of a small project.
- To acquire confidence to carry out major project in the subsequent semester.
- To develop a System/application/tool by using any Programming Language.
- To analyze and design the system using any UML tool.

Course Outcomes

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

- Analyze the problem statement.
- Model the content/architecture/interface/component for the system.
- Design the test cases to check the standards of the system.
- Prepare an environment to deploy the system.

PLAN AND DISTRIBUTION OF MARKS:

Week 1: Problem statement. **10 marks**

0th Review:

- Identification of title 5 marks
- Literature survey/scope 5 marks

Week 2,3,4: System Analysis **15 marks**

- System Requirement Specification(SRS)
- Software Engineering Paradigm applied
- Data models (like DFD), Control Flow diagrams, State Diagrams/Sequence diagrams, Entity Relationship Model, Class Diagrams/CRC Models/Collaboration Diagrams/Use-case Diagrams/Activity Diagrams depending upon your project requirements

1st Review:

- Methodology adopted 5 marks
- PPT presentation 5 marks
- Answers to queries 5 marks

Week 5,6,7: System Design **15 marks**

- Modularisation details
- Data integrity and constraints
- Database design, Procedural/Object Oriented Design
- User Interface Design
- Architecture design
- Component design
- Test Cases

2nd Review:

- Design aspects 5 marks
- PPT presentation 5 marks
- Answers to queries 5 marks

Week 8: System Implementation (coding) **15 marks**

Week 9: System testing **15 marks**

Week 10: Conclusion and Future Enhancements **10 marks**

Annexure

- User Manual
- Screens
- Bibliography and References used.

3rd Review:

- Results obtained 10 marks
- PPT presentation 10 marks
- Answers to queries 10 marks
- Project report 10 marks

Final Review:

- Presentation skills 05 marks
- Viva-Voce 15 marks

20 marks

II Year MCA – II Semester**CA261 : MAJOR PROJECT WORK**

Lectures	: ---	Sessional Marks	: 40
Practicals	: ---	Sem. End Exam Marks	: 60
Sem. End Exam Duration	: ---	Credits	: 10
Project Period	: Six months		

Prerequisites: CA253(Mini Project).

Course Objectives

- To analyze and design the system using any object-oriented modeling tool.
- To develop a system/application/tool in any domain.

Course Outcomes

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

- Understand all kinds of domains.
- Define a problem in any domain.
- Analyze, design and implement the system.
- Design the test cases to check the standards of the system.
- Prepare a report for system development.

Three Stages in Project adjudication:

- 1) Presentation of Problem Statement and Problem Approval by Guide.
- 2) Progress Approval by System Demonstration with results(Internal) - 40 Marks
- 3) Final Presentation with Documentation(External Project Viva-Voce)- 60 Marks

Note: Students are advised to follow the instructions of CA253.
