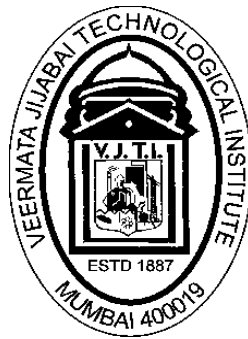


VEERMATA JIJABAI TECHNOLOGICAL INSTITUTE
(VJTI)
MATUNGA, MUMBAI 400 019

(Autonomous Institute affiliated to University of Mumbai)



Curriculum
(Scheme of Instruction & Evaluation and Course contents)
(Revision)

For
First Year
Of
Two Year Postgraduate Program Leading to
Master of Computer Applications (MCA)

Implemented from the batch admitted for FYMCA in 2020-2021

Vision and Mission statement of Institute:

Vision

To establish global leadership in the field of Technology and develop competent human resources for providing service to society

Mission

- To provide students with comprehensive knowledge of principles of engineering with a multi-disciplinary approach that is challenging
- To create an intellectually stimulating environment for research, scholarship, creativity, innovation and professional activity.
- To foster relationship with other leading institutes of learning and research, alumni and industries in order to contribute to National and International development.

Vision and Mission statement of MCA Department:

Vision:

To create a community of Critical Thinkers, Problem Solvers, Technological Experts who will be able to excel individually and collaboratively towards development of quality applications for betterment of business and society

Mission:

- Impart Quality Education to generate competent, skilled and Humane Manpower for computer application development and management.
- To include analytical skills for development of efficient, creative, innovative and user centric computer applications to support business and social causes.
- To create an environment for multifaceted development of students to make them industry ready in consultation with distinguished Alumni of the department
- To undertake collaborative projects which offer opportunities for long term interaction with academy and industry.
- Practice and promote high standards of professional ethics, transparency and accountability and ensure zero tolerance for lack of these core commitments.

Program Educational Objectives (PEOs)

1. Transcend in professional career and / or pursue higher education and research utilizing the knowledge gained in computational domain, mathematics, and management.
2. Ability to analyze real world problems, develop feasible and environmentally acceptable solutions to achieve peer recognition as an individual or in a team.
3. Work in multidisciplinary environment with ethical and sustainable computing perspectives, adaptable to the changing trends in technology and society by engaging in lifelong learning.
4. Identify opportunity to evolve as an entrepreneur and pursue the same for the benefit of individual and society.

Program Outcomes (PO)

1. Apply knowledge of computing fundamentals, computing specialization, mathematics, and domain knowledge appropriate for the computing specialization to the abstraction and conceptualization of computing models from defined problems and requirements.
2. Identify, formulate, research literature, and solve complex computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.
3. Design and evaluate solutions for complex computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.
4. Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations.
6. Understand and commit to professional ethics and cyber regulations, responsibilities, and norms of professional computing practice.
7. Recognize the need, and have the ability, to engage in independent learning for continual development as a computing professional.
8. Demonstrate knowledge and understanding of the computing and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
9. Communicate effectively with the computing community, and with society at large, about complex computing activities by being able to comprehend and write effective reports, design documentation, make effective presentations, and give and understand clear instructions.
10. Understand and assess societal, environmental, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to professional computing practice.
11. Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary environments.
12. Identify a timely opportunity and using innovation to pursue that opportunity to create value and wealth for the betterment of the individual and society at large.

Program Specific Outcomes (PSOs)

1. Analyze, Design, Test and Implement components, processes and solutions for specific application development using appropriate data modeling concepts and document the same.
2. Adapt and use appropriate modern software tools, resources and techniques to solve real world problems within the framework of constraints.
3. Apply concepts of networking and security to build and manage infrastructure to be

SEMESTER I

| Scheme of Instruction | | | | | Scheme of Evaluation | | | | | |
|-----------------------|-------------|---|-----------------------|---|----------------------|---------|----------|-----|-----|-----------|
| S. No | Course code | Course Title | L-T-P (Hours/week) | | | Credits | TA | MST | ESE | ESE hours |
| 1. | R5MC5011S | Management Information System | 3 | 0 | 0 | 3 | 20 | 20 | 60 | 3 |
| 2. | R5MC5012T | Software Engineering & Project Management | 3 | 0 | 0 | 3 | 20 | 20 | 60 | 3 |
| 3. | R5MC5013T | Operating System | 3 | 0 | 0 | 3 | 20 | 20 | 60 | 3 |
| 4. | R5MC5014S | Mathematical and Statistical Foundations 1 | 3 | 1 | 0 | 4 | 20 | 20 | 60 | 3 |
| 5. | R5MC5015S | Accounting and Finance(MOOC) | 3 | 0 | 0 | 3 | 60 | | 40 | 3 |
| 6. | R5MC5016L | Business English | 1 | 0 | 2 | 2 | 60 | | 40 | |
| 7. | R5MC5013P | Operating System Lab | 0 | 0 | 2 | 1 | 60 | | 40 | |
| 8. | R5MC5017L | Mobile Computing Lab | 0 | 0 | 2 | 1 | 60 | | 40 | |
| 9. | R5MC5018L | Web Technology Lab (Node Js, Angular Js, React, Flutter) | 0 | 0 | 2 | 1 | 60 | | 40 | |
| | R5MC5019D | Mini Project I(Based on SSAD, OOAD and User Experience Design Principles) | 0 | 0 | 2 | 1 | 100% CIE | | | |
| Total | | | 16 | 1 | 10 | 22 | | | | |

SEMESTER II

| Scheme of Instruction | | | | | | Scheme of Evaluation | | | | |
|-----------------------|-------------|---|-----------------------|---|----|----------------------|----|-----|-----|-----------|
| S. No | Course code | Course Title | L-T-P (Hours/week) | | | Credits | TA | MST | ESE | ESE hours |
| 1. | R5MC5021T | Data Mining | 3 | 0 | 0 | 3 | 20 | 20 | 60 | 3 |
| 2. | R5MC5022T | Design and Analysis of Algorithm | 3 | 0 | 0 | 3 | 20 | 20 | 60 | 3 |
| 3. | R5MC5023S | Mathematical and Statistical Foundation II | 3 | 1 | 0 | 4 | 20 | 20 | 60 | 3 |
| 4. | | Domain Elective I | 3 | 1 | 0 | 4 | 20 | 20 | 60 | |
| 5. | | Domain Elective II | 3 | 0 | 0 | 3 | 20 | 20 | 60 | |
| 6. | R5MC5027T | Professional Communication | 1 | 0 | 2 | 2 | 20 | 20 | 60 | |
| 7. | R5MC5021P | Data Mining Lab | 0 | 0 | 2 | 1 | 60 | | 40 | |
| 8. | R5MC5022P | Design and Analysis of Algorithm Lab | 0 | 0 | 2 | 1 | 60 | | 40 | |
| 9. | | Domain Elective II Lab | 0 | 0 | 2 | 1 | 60 | | 40 | |
| 10. | R5MC5025L | Java and Python Lab | 0 | 0 | 2 | 1 | 60 | | 40 | |
| 11 | R5MC5026D | Mini Project 2 (Based on RDBMS and User Experience Design Principles) | 0 | 0 | 2 | 1 | 60 | | 40 | |
| Total | | | 16 | 2 | 12 | 24 | | | | |

List of Electives:**Elective 1:**

| S. No | Course Code | Course Title |
|--------------|--------------------|-------------------------------------|
| 1. | R5MC5111S | Computer Graphics & Animation |
| 2. | R5MC5112S | Digital Forensics |
| 3. | R5MC5113S | Cloud Computing |
| 4. | R5MC5114S | Data Warehousing |
| 5. | R5MC5115S | Entrepreneurship Management and IPR |

Elective 2:

| S. No | Course Code | Course Title |
|--------------|--------------------|-----------------------------|
| 1. | R5MC5121T | Multimedia System |
| 2. | R5MC5122T | Image Processing |
| 3. | R5MC5123T | Software Design and Pattern |
| 4. | R5MC5124T | Ethical Hacking |
| 5. | R5MC5125T | Internet of Things |

| S. No | Course Code | Course Title |
|--------------|--------------------|---------------------------------|
| 1. | R5MC5121P | Multimedia System Lab |
| 2. | R5MC5122P | Image Processing Lab |
| 3. | R5MC5123P | Software Design and Pattern Lab |
| 4. | R5MC5124P | Ethical Hacking Lab |
| 5. | R5MC5125P | Internet of Things Lab |

| | | |
|------------------------------|--------------------------------------|--------------------------|
| <i>Programme Name</i> | <i>F.Y.M.C.A.</i> | <i>SEMESTER I</i> |
| Course Code | R5MC5011S | |
| Course Title | MANAGEMENT INFORMATION SYSTEM | |

PREREQUISITE

Basic Knowledge of managerial functions and organization

COURSE OUTCOMES

1. Describe various organizational structures, behaviors and its influence on MIS Design
2. Create reports for various subsystem in a organization based on their functionality and interrelationship
3. Explain the planning models and relevance of each in the current scenario at various levels of management.
4. Analyze the decision making requirements to create an appropriate decision support system.
5. Discuss advanced techniques in MIS.

COURSE CONTENTS

Introduction

Defining components of system, Data and Information, An introduction to Information systems, Information systems in organization and their capabilities, the MIS Pyramid, types of data requirements at each level, Foundation concepts: Business Applications, development and Management, Role and process of management, Functions of a manager, Methods of Management.

Types of Information Systems

Types of business Information System, Transaction Processing system, Management Reporting system, Decision Support system, Executive Information system, Office information system, Professional information system. Business Information system, Reports generated at each level and their dependencies.

IS planning

Strategic Planning Model, Management Control Model, Types of planning, Traditional Strategy making, Assumptions in traditional planning, Various Planning approaches: Traditional and Current scenario.

Functional subsystems

Marketing and Sales, Finance and Accounting, Production, Human Resources, Logistics and Inventory, Research and development. Identifying the data required and outputs for each functional subsystem and their dependencies on other functional subsystems

Decision support systems

Overview, Capabilities of DSS, DSS models: Scenario generation, Goal Seeking, DSS Components/ Architecture, DSS Classification, Building DSS .Group Decision Support System Characteristics of groups, Group decision making techniques, GDSS Architecture, GDSS Types/ Applications

Expert systems

Capabilities of ES, Architecture, Applications to Information Systems, Development and Maintenance of ES, Benefits and Limitations

Introduction to e-business

Models of E-business, Intranets/ Extranets, WWW, Effect on MIS

Ethical and societal issues

Ethical Issues, Societal issues, Dark Side of IT

Managing Green IT and smart cities

Introduction, What is Green IT?, Green practices in Hardware and Software industry, Green practices in IT use, Monitoring Value Chain Activities, Smart Cities, Facilities, Challenges for Managers

ICT for development of e-governance

ICT for development, Types of ICT Interventions, Examples of ICT for development projects, E-Governance concepts, E-Participation

REFERENCES:

1. Rahul, De. MIS: Managing Information Systems in Business, Government and Society, Wiley India Private Limited, Second Ed, 2018.
2. Barbara McNurlin et al, IS Management in practice, Pearson Education, 5th edition,
3. Zwass, Vladimir. Foundations of information systems. Irwin/McGraw Hill, 1997.
4. Laudon, Kenneth C., and Jane P. Laudon. "Management information systems: managing the digital firm." New Jersey 8 (2004).
5. W. S. Jawadekar, Management information Systems, Global Digital Enterprise Perspective, McGrawHill India, 5thed
6. James Obrien and George Maracus, Management information Systems McGrawHill India, 10thed,
7. Haag, Dawkins, Management information Systems for Information Age ,McGrawHill India, 6thed

| | | |
|-----------------------|--|-------------------|
| Programme Name | F.Y.M.C.A. | SEMESTER I |
| Course Code | R5MC5012T | |
| Course Title | SOFTWARE ENGINEERING AND PROJECT MANAGEMENT | |

PREREQUISITE

1. Should have sharp analytical, technical skills.
2. Should have Knowledge about Object oriented design model.
3. Should have Knowledge about Structured system analysis and design
4. Basic knowledge regarding business processes

COURSE OUTCOMES

1. Design a software system for a real worldproblem.
2. Evaluate the cost of the softwaresystem.
3. Design various testing strategy for the softwaresystem
4. Describe the roles and responsibility for resource requirements ofproject.
5. Justify the cost control measures of theproject.

COURSE CONTENTS

Introduction

Software Characteristics, Emergence of Software of Software Engineering. Software life cycle/process models, agile process, agile process models, Requirements Analysis and Specification: Requirements Gathering and Analysis, Software Requirement Specification (SRS), Project Size Estimation Metrics: Lines of Code (LOC), Function Point Metric, Feature Point Metric Software Scope, Software Estimation Techniques, Analytical Estimation Techniques, Risk Management.

Software design and software configuration management

Software Design: Overview of the Design Process, How to Characterize of a Design?, Cohesion and Coupling, Layered Arrangement of Modules, Approaches to Software Design, Necessity of Software Configuration Management baselines, SCM Process and SCL Configuration, Audit Version Control Source Code Control Systems (SCCS).

Fundamental of testing

Fundamentals of Test Process, Psychology of Testing, General Principles of Testing, Static Testing, Dynamic Testing, Black Box Testing, White Box Testing, Agile Methodology and Its Impact on testing, Test Levels, Test Organization, Test teams, tasks and Qualifications, Configuration Management, Specialized Testing: Performance, Load, Stress & Security Testing, Testing tools.

Project management

Introduction: Need, Goals, Evolution, Project environments, Planning fundamentals: Planning steps, Project master plan, Scope and work definition, Project organization structure and responsibilities, Project management system, Scheduling, Planning and scheduling charts, Cost estimating and budgeting

Project control

Control process, Control emphasis, Information monitoring, Internal and external project control, Traditional cost control, Cost accounting systems for project control, Performance analysis, Performance index monitoring, Variance limits, Controlling changes, Contract administration, Control problems

Project management information system

Functions of PMIS, Computer based tools, Computer –based PMIS, Representative Computer –based PMIS, Web based Project management, and Applying computer based PMS

Software quality and termination

Introduction, importance, software quality, ISO 9126, Software quality measures, External standards, Techniques to enhance software quality. Terminating the project, termination responsibilities, closing and contracts, Project extension

REFERENCES:

1. Softwareengineeringapractitioner’sApproach,RogerS.Pressman,SeventhEdition McGraw-Hill InternationalEdition.
2. Fundamentals of Software Engineering, Rajib Mall, Third Edition,PHI.
3. The Software Engineering: A Primer, Waman S Jawadekar, Tata McGraw-Hill,2008
4. Software Engineering, a Precise Approach, PankajJalote, WileyIndia, 2010.
5. The Software Engineering, Principles, and Practices, Deepak Jain, Oxford UniversityPress.
6. SoftwareEngineering1:Abstractionandmodeling,DinerBjorner,SpringerInternationaledition, 2006.
7. J.M.Nicholas, Project Management for Business and Technology 2nd Edition,PHI.
8. B.Hughes and Mike Cotterell, Software Project Management 4th Edition, Tata McGraw-Hill,2003
9. R.K.Wysocki, R.Beck Jr., D. B. Crane, John Wiley, Effective Project Management,1995.

| | | |
|-----------------------|-------------------------|-------------------|
| Programme Name | F.Y.M.C.A | SEMESTER-I |
| Course Code | R5MC5013T | |
| Course Title | OPERATING SYSTEM | |

PREREQUISITES

Good knowledge of C, Fundamentals of Machine, Bus architecture and memory management in computers

COURSE OUTCOMES

- 1 Summarize concurrency and processmanagement
- 2 Apply disk scheduling, memory management, CPU Schedulingalgorithms.
- 3 Explain the file structure, file system, directorysystem
- 4 Identify system Parameter, used to evaluate systemperformance
- 5 Describe the relation between tuning and Performanceparameter

COURSE CONTENTS

Introduction: Operating Systems

Introduction to Operating System, Role and Purpose of the operating System, Functionality of a typical operating system, Mainframe Systems, Desktop Systems, Multiprocessor Systems, Distributed Systems, Clustered Systems, Real - Time Systems, Handheld Systems, Feature Migration, Computing Environments.

Operating system structures: system structures

System Components (Compiler, Assembler, Linker, Loader, OS, I/O manager), Structuring methods (monolithic, layered, modular, micro-kernel models), Operating – System Services, Abstractions, processes and resources, System Calls, System Programs, System Structure, Virtual Machines, System Design and Implementation, System Generation. Design issues (efficiency, robustness, flexibility, portability, security, compatibility), Influences of security, networking, multimedia, windowingsystems.

Process management

Concept of Process and threads, Process States, Process Description, Process Control, Context switching, Execution of the Operating System, Security Issues, Processes and Threads, Symmetric Multiprocessing(SMP), Microkernels, CPU Scheduler and Scheduling.

Concurrency: mutual execution and synchronization

The need for interprocess synchronization, Principles of Concurrency, Race condition, Mutual exclusion, semaphores, hardware support for mutual exclusion, Queuing implementation of semaphores, classical problems in concurrent programming, critical region, monitors and deadlocks, Message Passing, Readers/Writes Problem, Producer/Consumer problem

Deadlock and starvation

Principles of Deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, an Integrated Deadlock Strategy, Deadlock Recovery, Dining Philosophers Problem

Memory management

Memory Partitioning, Swapping, Contiguous Memory Allocation, Paging, Paging with segmentation, Segmentation, Segmentation with Paging, Demand Paging, Process Creation, Page Replacement Algorithms, Allocation of Frames and relevant algorithms, Thrashing.

File system- interface and implementation

File Concepts (data, metadata, operations, organization, buffering, sequential, non-sequential), Access Methods, Directory Content Structure, File System partitioning, mounting / unmounting, File sharing, Protection, File System structure, File system Implementation, Directory Implementation, Allocation methods, free-space management.

Secondary storage, security and protection

Disk Structure, Disk Scheduling, Disk Management, The Security Problem, User Authentication, Program Threats, System Threats. Security methods and devices

Performance management, monitoring and evaluation

Introduction, important trends affecting performance issues, why performance monitoring and evaluation are needed, performance measures, evaluation techniques, Bottlenecks and saturation, feedback loops.

Case studies

MS DOS MS Windows, Linux (UNIX) Operating Systems

REFERENCES:

1. William Stallings, Operating System, Internals and Design Principles, Prentice Hall, 7th edition, 2011.
2. Silberschatz, Galvin, Wiley India, Operating System Concepts, John Wiley & Sons, 7th Edition, 2006.
3. Maurice Bach, The design of the UNIX Operating System, Pearson, 1st Edition, 1986.
Andrew Tanenbaum Modern Operating Systems, PHI, 3rd Edition, 2009.

| | | |
|-----------------------|-----------------------------|-------------------|
| Programme Name | F.Y. M.C.A. | SEMESTER-I |
| Course Code | R5MC5013P | |
| Course Title | OPERATING SYSTEM LAB | |

PREREQUISITES

Good knowledge of C

COURSE OUTCOMES

1. Apply CPU scheduling algorithms.
2. Apply Memory management algorithms.
3. Evaluate Disk-scheduling algorithms.
4. Simulate resource management in OS.

TITLE OF EXPERIMENTS

1. Implement the following CPU scheduling.
 - A) FCFS B) SJF (Preemptive and Non Preemptive)
2. Implement the following CPU scheduling
 - a) Round Robin b) Priority
3. Implement mutual exclusion using Semaphores
4. Implement Bankers algorithm for Dead Lock Avoidance
5. Implement the all page replacement algorithms a) FIFO b) LRU c) LFU
6. Implementation of Disk-scheduling algorithms
 - a)FCFS b)SSTF (shortest seek time first) algorithm c)SCAN scheduling
7. Implement the following Disk-scheduling algorithms
 - a) C-SCAN scheduling b)LOOK Scheduling c)C-LOOK scheduling

ASSESSMENT

The distribution of marks for ESE practical exam is as follows:

- **Practical test \Quiz: 20 marks**
- **Viva-Voice : 20 marks**

| | | |
|-----------------------|---|-------------------|
| Programme Name | M.C.A | SEMESTER-I |
| Course Code | R5MC5014S | |
| Course Title | MATHEMATICAL AND STATISTICAL FOUNDATIONS I | |

PREREQUISITES

Basic concepts of calculus - integration and differentiation, Univariate and bi-variate Analysis, Knowledge of R programming.

COURSE OUTCOMES

1. Analyze data using probability and statistical techniques.
2. Solve a system of linear equations and perform dimensionality reduction using singular value decomposition, principal component analysis of given data.
3. Apply analytic geometry concepts for measuring length, distances, calculate gradient and perform geometric transformations on vectors.
4. Find trends in given data and make forecasts.
5. Apply the concepts of graphs and trees to model real-world problems.

COURSE CONTENTS

Probability

Introduction to probability, Conditional Probability, Bayes Theorem, Random Variables, Probability Distributions: Bernoulli Distribution, Uniform Distribution, Binomial Distribution, Normal Distribution, Poisson Distribution, Geometric Distribution, Exponential Distribution.

Statistics

Introduction to Statistics: Population vs Sample, Dependent and Independent variables, Categorical and Continuous Variables, Central Tendency, Variance, Covariance, Correlation, Central Limit Theorem, Descriptive statistics: Multivariate Analysis (Multivariate scatter plot, grouped box-plot, heat-map), Function Models (linear, exponential, logistic)

Inferential Statistics: Sampling Distributions, Hypothesis Testing (t-testing), normality, Confidence Intervals, Estimation.

Regression: Simple Linear Regression, Multiple Linear Regression, Nonlinear Regression

Linear Algebra

Systems of Linear Equations, Matrices, Multiplying Matrices, Identity and Inverse Matrices, Special Kinds of Matrices (Diagonal, Symmetric & Orthogonal matrices), Solving Systems of Linear Equations, Linear Independence

Analytic Geometry

Norms, Inner Products, Lengths and Distances, Angles and Orthogonality, Orthonormal Basis, Inner Product of Functions, Orthogonal Projections, Rotations.

Matrix Decomposition

Determinant and Trace, Eigenvalues and Eigenvectors, Eigen decomposition and Diagonalization, Singular Value Decomposition, Principal Component Analysis.

Time series analysis

Trends and seasonal variations, exponential smoothing, forecasting techniques with focus on predictions and error analysis of results

Graph Theory

Definition of a graph, Path, circuits, reachability, connectedness, Matrix representation of graphs, List structures, and graphs, General tree, Binary tree, Tree traversals

RECOMMENDED READING

1. Deisenroth, Marc Peter. Mathematics for Machine Learning, Cambridge University Press, 2020.
2. Gilbert Strang, Introduction to Linear Algebra, Wellesley-Cambridge Press, Fifth edition, 2016.
3. Gilbert Strang, Calculus Online Textbook, Massachusetts Institute of Technology, 2005.
4. Trivedi, Kishor Shridharbhai. Probability and Statistics with Reliability, Queuing, and Computer Science Applications. Wiley, 2016.
5. Tremblay, Manohar, Discrete Mathematical Structures, Tata McGraw-Hill Education, 2001.

| | | |
|-----------------------|-----------------------------|-------------------|
| Programme Name | F.Y. M.C.A. | SEMESTER-I |
| Course Code | R5MC5017L | |
| Course Title | MOBILE COMPUTING LAB | |

COURSE OUTCOMES

1. Familiarity with basics of cellular networks with advancement in higher generation.
2. Developing Mobile application using Android/other.

TITLE OF EXPERIMENTS

1. Compare different mobile development platforms including cross platform development platforms.
2. Detailed comparative analysis of 1G, 2G, 3G, 4G and 5G.
3. Use various controls like Edit View, buttons, radio buttons, checkboxes,AutoCompleteTextView, Image Button,and Toggle Button on Mobile to develop UI using Android /other.
4. Design a simple calculator using Android/other as a separate module of previous.
5. Create a simple temperature converter application using Android/other as a separate module of previous.
6. Write a Program to generate Calendar using Android/other.
7. Design a simple to-do list using Android/other.
8. Write a Program to insert and display data from database using Android/other.
9. Write a Program for simple quiz competition Android/other.
10. Write a Program to demonstrate simple Animation Android/other.

ASSESSMENT

The distribution of marks for ESE practical exam is as follows:

- **Practical test / Quiz: 20 marks**
- **Viva-Voice : 20 marks**

| | | |
|-----------------------|---------------------------|-------------------|
| Programme Name | F.Y.M.C.A | SEMESTER-I |
| Course Code | R5MC5018L | |
| Course Title | WEB TECHNOLOGY LAB | |

PREREQUISITES

Working knowledge of HTML and JavaScript

COURSE OUTCOMES

1. Design Dynamic web pages using various components to meet the specified needs with appropriate tools.
2. Understand the JavaScript and technical concepts behind Node JS
3. Reduce the amount of code you write to build rich user interface applications.

TITLE OF EXPERIMENT

1. **Design an experiment to demonstrate Node JS.**
 - a) Introduction to Node JS
 - b) Setup Dev. Environment
 - c) Node JS Modules
 - d) Node Package Manager
 - e) Creating Web server
 - f) File System
 - g) Debugging Node JS Application
 - h) Events
 - i) Express.JS
 - j) Database connectivity and Template Engines
2. **Design an experiment to demonstrate Angular JS.**
 - a) Introduction
 - b) Expressions and Data Binding
 - c) Working with Directives and Controllers
 - d) Filters
 - e) Forms
 - f) Modules and Services
3. Demonstrate React Framework.
4. Study the Flutter environment.

ASSESSMENT

The distribution of marks for ESE practical exam is as follows:

- **Viva-Voice : 20 Marks**
- **Practical test /Quiz : 20 Marks**

| | | |
|-----------------------|-----------------------|-------------------|
| Programme Name | F.Y. M.C.A. | SEMESTER-I |
| Course Code | R5MC5019D | |
| Course Title | MINI PROJECT 1 | |

PREREQUISITES

Knowledge of any programming language for developing the project and excellent communication skill will be and added advantage.

COURSE OUTCOMES

1. Formalize the requirement gathering document for givensystem.
2. Analyze the design the given system using Structural/ Staticmodels.
3. Analyze the design the given system using Behavioral / Dynamicmodels.
4. Create Behavioral diagrams using Use Case, Structuraldiagrams.

TITLE OF EXPERIMENT

1. Introduction to UML:UML Overview, The Nature and purpose ofModels
2. Modeling Requirements(Use Cases):Capturing a System Requirement, Use Case Relationships, Use Case OverviewDiagrams
3. Modeling System Workflows(Activity Diagrams):Activity Diagram Essentials, Activities and Actions, Decisions and Merges, Doing Multiple Tasks at the Same Time, Time Events, Objects, Sending and Receiving Signals, Starting an Activity, Ending Activities and Flows, Partitions (or Swim lanes), Managing Complex ActivityDiagrams
4. Modeling a System's Logical Structure: Introducing Classes and Class Diagrams, Advanced Class Diagrams What is a Class?, Getting Started with Classes in UML, Visibility, Class State: Attributes, Class Behavior: Operations, Static Parts of Your Classes, Class Relationships, Constraints, Abstract Classes, Interfaces,Templates
5. Bringing Your Classes to Life: Object Diagrams Object Instances, Links, Binding Class Templates
6. Modeling Ordered Interactions: Sequence Diagrams, Participants in a Sequence Diagram, Time, Events, Signals, and Messages, Activation Bars, Nested Messages, Message Arrows, Bringing a Use Case to Life with a Sequence Diagram, Managing Complex Interactions with SequenceFragments
7. Focusing on Interaction Links: Communication Diagrams, Focusing on Interaction Timing: Timing Diagrams Participants, Links, and Messages, Fleshing out an Interaction with a Communication Diagrams, Communication Diagrams Versus Sequence Diagrams What Do Timing Diagrams Look Like?, Building a Timing Diagram from a Sequence Diagram, Applying Participants to a Timing Diagram, States, Time, A Participant's State-Line, Events and Messages, TimingConstraints. Completing the Interaction Picture: Interaction Overview Diagrams.
8. Managing and Reusing Your System's Parts (Component Diagrams):What is a Component?, A Basic Component in UML, Provided and Required Interfaces of a Component, Showing Components Working Together, Classes that Realize a Component, Ports and Internal Structure, Black-Box and White-BoxComponent Views
9. Modeling an Object's State: State Machine Diagrams Essentials, States, Transitions, States in Software, Advanced State Behavior, Composite States,Signals
10. Modeling Your Deployed System(Deployment Diagrams):Deploying a Simple System, Deployed Software: Artifacts, What Is a Node?, Hardware and Execution Environment Nodes, Communication Between Nodes, Deployment Specifications, When to Use a Deployment

Diagram

11. UML tools and techniques for web-based/object oriented: Applications UML Tools, Different UML Notations for Webapplication
12. Creation of documentation such as SRS, SDS from UML diagrams. Generation of code from UML model. Basic Concept, Generating by Templates, Using Batches, Installing and Uninstalling Templates
13. Mini Project: A Mini – Project based on Java Programming and UML using an integrated approach. (Maximum Three students in a Group).

| | | |
|-----------------------|-------------------------|-------------------|
| Programme Name | F.Y. M.C.A. | SEMESTER-I |
| Course Code | R5MC5016L | |
| Course Title | BUSINESS ENGLISH | |

PREREQUISITE

Basic English

COURSE OUTCOMES

1. Acquire basic proficiency in English grammar and vocabulary.
2. Develop enhanced writing skills required for technical communication.
3. Demonstrate skills in delivery of formal speeches and presentations.

COURSE CONTENTS

Vocabulary Building

The concept of Word Formation. Root words from foreign languages and their use in English. Acquaintance with prefixes and suffixes from foreign languages in English to form derivatives. Technical vocabulary specific to engineering fields.

Basic Writing Skills

(This unit involves interactive practice sessions in Language Lab) Sentence Structures. Use of phrases and clauses in sentences. Importance of proper punctuation. Creating coherence. Organizing principles of paragraphs in documents

Identifying Common Errors in Writing

Subject-verb agreement. Noun-pronoun agreement. Misplaced modifiers. Articles
Prepositions

Writing Practices

Summarization, email, business letters

Speaking skills

Overcoming fear, organizing and delivering speeches.

Text Books

1. Practical English Usage. Michael Swan. OUP. 1995.
2. Remedial English Grammar. F.T. Wood. Macmillan. 2007

Recommended Reading

1. On Writing Well. William Zinsser. Harper Resource Book. 2001

SEMESTER II

| Scheme of Instruction | | | | | Scheme of Evaluation | | | | | |
|-----------------------|-------------|---|---------------------------|---|----------------------|---------|----|-----|-----|-----------|
| S. No | Course Code | Course Title | L-T-P (Hours/ week) | | | Credits | TA | MST | ESE | ESE hours |
| 1. | R5MC5021T | Data Mining | 3 | 0 | 0 | 3 | 20 | 20 | 60 | |
| 2. | R5MC5022T | Design and Analysis of Algorithm | 3 | 0 | 0 | 3 | 20 | 20 | 60 | |
| 3. | R5MC5023S | Mathematical and Statistical Foundation II | 3 | 1 | 0 | 4 | 20 | 20 | 60 | |
| 4. | | Domain Elective I | 3 | 1 | 0 | 4 | 20 | 20 | 60 | |
| 5. | | Domain Elective II | 3 | 0 | 0 | 3 | 20 | 20 | 60 | |
| 6. | R5MC5027L | Professional Communication | 1 | 0 | 2 | 2 | 40 | | 60 | |
| 7. | R5MC5021P | Data Mining Lab | 0 | 0 | 2 | 1 | 60 | | 40 | |
| 8. | R5MC5022P | Design and Analysis of Algorithm Lab | 0 | 0 | 2 | 1 | 60 | | 40 | |
| 9. | | Domain Elective II Lab | 0 | 0 | 2 | 1 | 60 | | 40 | |
| 10. | R5MC5025L | Java and Python Lab | 0 | 0 | 2 | 1 | | | | |
| 11 | R5MC5026D | Mini Project 2 (Based on RDBMS and User Experience Design Principles) | 0 | 0 | 2 | 1 | 60 | | 40 | |
| Total | | | 16 | 2 | 12 | 24 | | | | |

| | | |
|-----------------------|--------------------|--------------------|
| Programme Name | F.Y.M.C.A. | SEMESTER II |
| Course Code | R5MC5021T | |
| Course Title | DATA MINING | |

PREREQUISITE

Knowledge of Databases is required for this course, and programming language like Java, R or Python is desired for this course

COURSE OUTCOMES

1. Describe and demonstrate the basic concept of data mining and data visualization.
2. Preprocess the data for given real – world problem to choose the relevant models and algorithms to apply.
3. Write simple DMQL queries.
4. Solve the data mining problems using appropriate techniques of association rule mining, classification, clustering and web mining.
5. Discuss various aspects of advanced data mining techniques.

COURSE CONTENTS

Introduction to Data Mining

Introduction: Fundamentals of data mining, Data mining Functionalities, Classification of Data Mining Systems, Data Mining Motivation, Challenges, Data Mining Tasks, Knowledge Discovery : KDD Process, KDD vs data mining, Data Mining Task Primitives, Integration of data mining system with a database or data warehouse system, Major issues in Data Mining.

Understanding Data and Data visualization

Types of data, Data cleaning and ensuring Quality, Data Preprocessing:- Aggregation, Sampling, Dimensionality Reduction, Feature Creation, Discretization and Binarization ,Measures of Similarity and Dissimilarity, Data Exploration, Summary Statistics, Visualization.

Classification and Prediction

Issues regarding Classification and Prediction, General approach to solving a Classification Problem, Decision tree Induction, Rule based classification, Nearest-Neighbor classifier, Naïve Bayesian Classifier, Other classification methods for improvement of classification accuracy, Evaluating Accuracy of a classifier or predictor, Other classification methods, Prediction, Accuracy and Error measures, Ensemble methods

Cluster Analysis

Basic Concepts of Cluster Analysis, Different Types of Clustering, Partitioning methods – k-means, k-medoids, Agglomerative Hierarchical Clustering, Evaluation of Clustering, Graph-Based Clustering, Density-Based Methods, Outlier Analysis.

Association Rule Mining

Basic concepts, Frequent Itemset Generation, Rule Generation, Alternative Methods for Generating Frequent Itemsets, FP-Growth Algorithm, Handling Categorical Attributes, Handling Concept Hierarchy, Sequential Patterns, Associations and Correlations: Basic concepts, mining various kinds of association rules, from association mining to correlation analysis, Graph mining.

Mining Objects, Spatial, Multimedia, Text and Web Data Mining

Multidimensional Analysis and Descriptive Mining of Complex, Data Objects, Web Content Mining, Web Structure Mining, Web Usage Mining, Spatial Data Overview, Spatial Data Mining primitives, Generalization/Specialization, Spatial Rules, Spatial Classification, Spatial clustering, Modelling Temporal Events, Time series, Pattern Detection, Sequences, Temporal Association rules, Applications and Trends in Data Mining, Data Mining applications.

Data Mining Primitives

Languages, Data mining primitives, Query language.

RECOMMENDED READING

1. Pang-Ning Tan, Michael Steinbach, Vipin Kumar, Introduction To Data Mining, Pearson 2nd Edition.
2. A Programmer's Guide to Data Mining, Ron Zacharski, <http://guidetodatamining.com/>
3. Jaiwei Han and Micheline Kamber, Third Edition, Morgan Kaufman , Data Mining Concepts and Techniques
4. Dunham, Margaret H. Data mining: Introductory and advanced topics. Pearson Education India, 2006.
5. ReemaTheraja , Data Warehousing, Oxford Higher Education press

| | | |
|-----------------------|------------------------|--------------------|
| Programme Name | F.Y.M.C.A. | SEMESTER II |
| Course Code | R5MC5021P | |
| Course Title | DATA MINING LAB | |

PREREQUISITE

A good knowledge of any one programming language like Java, introductory knowledge of R or Python.

COURSE OUTCOMES:

1. Analyze classical models, algorithms in data mining
2. Preprocess the data for given real – world problem to choose the relevant models and algorithms to apply
3. Write simple DMQL queries
4. Implement classification, clustering and web mining algorithms.

TITLE OF EXPERIMENT

1. Apply concepts of Data cleaning and preprocessing on real world datasets.
2. Apply decision tree algorithm and Naive Bayes algorithm to a suitable dataset and interpret the results. Compare the performance on the same dataset with different technique.
3. Implement any 2 ensemble methods on a suitable dataset and interpret the results. Compare the performance on the same dataset with different technique.
4. Write 10 DMQL queries for suitable datasets.
5. Clustering algorithm
 - a. Apply any one distance based algorithm and interpret the result.
 - b. Apply any two hierarchical algorithms and interpret the result.
 - c. Apply DB Scan clustering algorithm to data and interpret the result.

Compare the performance on the same dataset with different technique.
6. Association mining – frequent item set mining and implement frequent pattern growth algorithm.
7. Apply classification and association to weblog or suitable web data.

ASSESSMENT

The distribution of marks for ESE practical exam is as follows:

- **Viva-Voice : 20 Marks**
- **Practical test /Quiz : 20 Marks**

| | | |
|-----------------------|---|--------------------|
| Programme Name | F.Y.M.C.A. | SEMESTER II |
| Course Code | R5MC5022T | |
| Course Title | DESIGN AND ANALYSIS OF ALGORITHM | |

PREREQUISITE

Basic understanding of data structures and algorithm analysis.

COURSE OUTCOMES

1. Distinguish between different algorithm designs techniques.
2. Apply the algorithm analysis techniques.
3. Evaluate the limitations of Algorithmic power.
4. Critically analyze the efficiency of alternative algorithmic solutions for the same problem.

COURSE CONTENTS

Introduction

Notion of an Algorithm , Fundamentals of Algorithmic Problem Solving , Important Problem Type, Fundamentals of the Analysis of Algorithmic Efficiency , Asymptotic Notations and Basic Efficiency Classes-Informal Introduction , O-notation , -notation -notation , Mathematical Analysis of Non recursive Algorithms, Mathematical Analysis of Recursive Algorithms, Empirical Analysis of Algorithms , Algorithm Visualization

Brute force and divide-and-conquer

Brute Force , Computing an, String Matching , Closest-Pair and Convex-Hull Problems ,Exhaustive Search , Travelling Salesman Problem , Knapsack Problem , Assignment problem, Divide and Conquer Methodology, Binary Search , Merge sort , Quick sort , Heap Sort ,Multiplication of Large Integers , Closest-Pair and Convex – Hull Problems.

Dynamic programming and greedy technique

Dynamic programming, Principle of optimality, Coin changing problem, Computing a Binomial Coefficient, Floyd’s algorithm, Multi stage graph, Optimal Binary Search Trees, Knapsack Problem and Memory functions. Greedy Technique, Container loading problem, Prim’s algorithm and Kruskal’s Algorithm, 0/1 Knapsack problem, Optimal Merge pattern, Huffman Trees.

Iterative improvement

The Simplex Method , The Maximum-Flow Problem , Maximum Matching in Bipartite Graphs, Stable marriage Problem.

Limitations of algorithm power

Lower – Bound Arguments – P, NP NP- Complete and NP Hard Problems. Backtracking – n-Queen problem, Hamiltonian Circuit Problem, Subset Sum Problem. Branch and Bound, LIFO Search and FIFO search, Assignment problem, Knapsack Problem, Travelling Salesman Problem, Approximation Algorithms for NP-Hard Problems

TEXT BOOKS:

1. Anany Levitin, —Introduction to the Design and Analysis of Algorithms, Third Edition, Pearson Education, 2012.
2. Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, Computer Algorithms/ C++, Second Edition, Universities Press, 2007.

REFERENCES:

1. Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest and Clifford Stein, —Introduction to Algorithms, Third Edition, PHI Learning Private Limited, 2012.
2. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, —Data Structures and Algorithms, Pearson Education, Reprint 2006.
3. Harsh Bhasin, —Algorithms Design and Analysis, Oxford university press, 2016. 4. S. Sridhar, —Design and Analysis of Algorithms, Oxford university press, 2014. 5. <http://nptel.ac.in/>

| | | |
|-----------------------|--|--------------------|
| Programme Name | F.Y. M.C.A. | SEMESTER II |
| Course Code | R5MC5022P | |
| Course Title | DESIGN AND ANALYSIS OF ALGORITHMS LAB | |

PREREQUISITES

1. C/C++/Java programming, data structures

COURSE OUTCOME

1. Implement divide & conquer method
2. Apply greedy method to solve the given problem.
3. Apply dynamic programming method to given problem.
4. Solve problems using backtracking approach.
5. Solve problems using branch and bound approach.

TITLE OF EXPERIMENT

1. Implement the solution to the following using divide and conquer approach:
 - A. Sorting given data using merge sort.
 - B. Counting inversions in given data.
 - C. Multiplying large integers.
2. Implement the solution to the following using greedy approach:
 - A. 0/1 Knapsack problem
- B. Finding the shortest path in a graph. (Dijkstra's algorithm)
3. Implement the solutions to the following using dynamic programming approach:
 - A. Subset sum problem
 - B. Fractional knapsack problem
 - C. Finding the shortest path in a graph with negative edge weights. (Bellman-Ford's algorithm)
4. Write a program to solve 0/1 knapsack problem using backtracking approach.
5. Write a program to solve 0/1 knapsack problem using branch and bound approach.

ASSESSMENT

The distribution of marks for ESE practical exam is as follows:

- **Viva-Voice : 20 Marks**
- **Practical test /Quiz : 20 Marks**

| | | |
|-----------------------|---|--------------------|
| Programme Name | F.Y. M.C.A. | SEMESTER-II |
| Course Code | R5MC5023S | |
| Course Title | MATHEMATICAL AND STATISTICAL FOUNDATIONS 2 | |

COURSE OUTCOMES

1. Understand system based on mathematical models in abstract algebra
2. Solve the regular expressions and find equivalent regular languages
3. Apply continuous optimization using gradient descent.
4. Estimate density with Gaussian mixture models
5. Apply support vector machine algorithm to find a hyperplane in an N-dimensional space

COURSE CONTENTS

Abstract Algebra and Number Theory

Modular Inverse, Extended Euclid Algorithm, Fermat's Little Theorem, Euler Phi-Function, Euler's theorem, Integer factorization problem, Discrete logarithm problem, Group theory, Ring, Field, Primarily Testing, ElGamal Cryptosystem, Elliptic Curve over the Reals, Elliptic curve Modulo a Prime, Chinese remainder theorem, Jacobi Symbol, Birthday Paradox

Finite Automata and Regular Languages

Deterministic finite automata, Regular operations, Nondeterministic finite automata, Equivalence of DFAs and NFAs, Closure under the regular operations, Regular expressions, Equivalence of regular expressions and regular languages, The pumping lemma and nonregular languages, Context-Free Languages, Context-free grammars, Chomsky normal form, Pushdown automata, Equivalence of pushdown automata and context-free grammars, The pumping lemma for context-free languages

Vector calculus

Partial Differentiation and Gradients, Gradients of Vector-Valued Functions, Gradients of Matrices, Identities for Computing Gradients, Back propagation and Automatic Differentiation, Higher-Order Derivatives.

Continuous optimization

Optimization Using Gradient Descent, Constrained Optimization and Lagrange Multipliers, Convex Optimization.

Density Estimation with Gaussian Mixture Models

Gaussian Mixture Model, Parameter Learning via Maximum Likelihood, EM Algorithm, Latent-Variable Perspective

Classification with Support Vector Machines

Separating Hyperplanes, Primal Support Vector Machine, Dual Support Vector Machine, Kernels, Numerical Solution

REFERENCES:

1. Deisenroth, Marc Peter, A. Aldo Faisal, and Cheng Soon Ong. *Mathematics for machine learning*. Cambridge University Press, 2020.
2. Dummit, David Steven, and Richard M. Foote. *Abstract algebra*. Vol. 3. Hoboken: Wiley, 2004.
3. Kraft, James, and Lawrence Washington. *An introduction to number theory with cryptography*. CRC Press, 2018.
4. Maheshwari, Anil, and MichielSmid. "Introduction to Theory of Computation." *School of Computer Science Carleton University Ottawa Canada*. 2014.
5. Kakde, O. G. *Theory of Computation*. Firewall Media, 2007.

| | | |
|-----------------------|--|--------------------|
| Programme Name | M.C.A | SEMESTER-II |
| Course Code | R5MC5111S | |
| Course Title | COMPUTER GRAPHICS & ANIMATION(Elective 1) | |

PREREQUISITES

A good knowledge of C

COURSE OUTCOMES

1. Discuss various algorithms to draw lines, circles and ellipses.
2. Apply transformation techniques to 2D and 3D objects.
3. Apply concepts of graphics for animation on real world scenario.
4. Describe interactive methods.

COURSE CONTENTS

Overview of Graphics Systems

Introduction to various Graphic devices, Raster scan display processor, Graphics software, Coordinate representations, Graphics functions, software standards

Output Primitives:

Points and lines, Line drawing algorithms, Loading the frame buffer, Line Function, Circle generating algorithms, Other curves, Parallel curve algorithms, Curve functions, Pixel addressing, Filled area Primitives, Fill area functions., attributes of output primitives.

Two Dimensional Geometric Transformations

Basic Transformations, Matrix Representation, Composite Transformations, Reflection, Shear, Transformation between Coordinate Systems.

Two Dimensional Viewing

The viewing Pipeline, Viewing Coordinate Reference Frame, Two dimensional viewing Functions, Point clipping, Line Clipping, Polygon Clipping, Curve Clipping, Text Clipping, Exterior Clipping.

Structures and Hierarchical Modeling

Structure Concept, Editing Structures, Basic Modeling Concepts, Hierarchical Modeling.

GUI and Interactive Input Methods

Input of Graphical Data, Input Functions, Interactive Picture Construction Techniques, Virtual Reality environments.

Three Dimensional Concepts

Parallel Projection, perspective projection, Depth Cueing, Visible Line and surface Identification, Surface Rendering, and Three Dimensional and cutaway views.

Three Dimensional object representations

Bezier curves and surfaces, B-spline curves and surfaces.

Visible surface Detection Method

Back face detection, Depth Buffer Method, Depth sorting method, OCTREE Method

Illumination model and surface Rendering Methods

Light sources, Basic illumination model, Halftone patterns and Dithering Techniques. Polygon Rendering Methods.

Computer Animation

Design of Animation sequences, General computer Animation function, Raster Animations, Computer animation languages, key frame system, morphing, simulation acceleration, Motion specification..

REFERENCES

1. Hearn, Donald. "M. Pauline Baker Computer Graphics." (1997).
2. NarendraSinha, ArunUdai, Computer graphics, TMG

| | | |
|-----------------------|--------------------------------------|--------------------|
| Programme Name | M.C.A | SEMESTER-II |
| Course Code | R5MC5112S | |
| Course Title | DIGITAL FORENSIC (Elective 1) | |

PREREQUISITES

1. Basic networking concepts
2. Basic knowledge of Cryptography and information security
3. Basic knowledge of programming

COURSE OUTCOMES

1. Describe basic concepts and terminologies in digital forensic and ethical hacking.
2. Identify methods for data recovery.
3. Analyze various computer forensic tools
4. Apply technique to protect digital evidence
5. Identify the various cyber laws and their context

COURSE CONTENTS

Introduction to digital forensic

Introduction, Reasons for cybercrime, Motive behind cybercrime, Cyber criminals Modes and manner of committing cybercrimes, Classification of cybercrimes, Preventing Cyber Crimes, Code Hacking- Input Validation, Virus and Worms, Steganography, DoS, DDoS Attacks, SQL Injection Buffer Overflow Attacks, Cross Side Scripting, Ethical hacking of operating Systems (Windows, Linux and Mac OS), Ethical hacking of web, email and mobile Phones.

Digital evidence collection

Evidence collection- locating evidences in the file system, defining special evidence, understanding technical complexities of digital evidence, Search and seizure of digital evidence, acquisition and safekeeping of digital evidence, various process involved in the safekeeping, disk image.

Special forensic investigation

Email Forensics Investigation, Data storage Forensics, Investigation of mobile devices, Forensics, and investigation of Wi-Fi Environment.

Cyber security laws

The Legal Perspectives Why do we need Cyber law: The Indian Context, Positive and Weak areas of ITA 2000, Information Security Standard compliances: SOX, GLBA, HIPAA, ISO, FISMA, NERC, PCI-DSS, International Laws: E-Sign, CIPA and COPPA.

Open source tools

Tools, Security and Forensics Software's Tools: FTK (Forensics Tool Kit), EnCase, TCT (The Coroners Toolkit), TSK (The Sleuth Kit).

Security software

Antivirus, Email Security, Identify and Access Management, Incidence response policies, Incidence reporting, Forensics & Intrusion Detection, and Prevention. Forensics Software: Password Cracking Tool, Open Source Tool, Mobile Devices Tool (PDA/ Cell phone), Large Storage Analysis.

Case study and scenarios:

IP Thefts, Corporate Frauds, Digital Frauds, Cyber Crimes, Cyber Porn, Cyber Stalking, Consumer and credit Card Fraud, Online and Digital Fraud- Phishing Attacks, Spare Attack and other Incidents.

RECOMMENDED READING

1. Marjie T. Britz, Computer Forensics and Cyber Crime an Introduction, Pearson, 3rd Edition
2. Benild Joseph, Digital Crime Investigation: Handbook for Cyber Crime Investigators
3. Britz, Computer Forensics and Cyber Crime: An Introduction, 2e
4. v. Jason Luttgens, Matthew Pepe, Kevin Mandia, "Incident Response and computer forensics", 3rd Edition Tata McGraw Hill, 2014.
5. Nilakshi Jain, DhananjayKalbande, "Digital Forensic: The fascinating world of Digital Evidences" Wiley India Pvt Ltd 2017.
6. Cory Altheide, Harlan Carvey" Digital forensics with open source tools "Syngress Publishing, Inc. 2011.
7. Chris McNab, Network Security Assessment, By O'Reily

| | | |
|-----------------------|-------------------------------------|--------------------|
| Programme Name | F.Y. M.C.A. | SEMESTER II |
| Course Code | R5MC5113S | |
| Course Title | CLOUD COMPUTING (Elective 1) | |

PREREQUISITES

1. Basic networking concepts
2. Knowledge of programming language
3. Basic knowledge of information security

COURSE OUTCOMES

1. Explain the basic concepts and terminologies in cloud computing and its applications.
2. Distinguish between different types of architectures and services in the cloud computing.
3. Describe the management of cloud services.
4. Analyze different techniques of cloud security.
5. Describe the characteristics and different components of different cloud implementations in the real world.

COURSE CONTENTS

Introduction to cloud computing:

Overview of Computing Paradigm: Grid Computing, Cluster Computing, Distributed Computing, Utility Computing, and Cloud Computing; NIST Model of Cloud Computing: Definition, Essential Characteristics, Service Models, and Deployment Models; Challenges and Risks; Overview of the various Cloud Service Providers and Applications.

Infrastructure as a service (IaaS):

IaaS Definition; Introduction to Virtualization, Characteristics of Virtualized Environments, Taxonomy of Virtualization Techniques; Hypervisors, Virtual Machine (VM): Provisioning and Migration, Resource Virtualization: Server, Storage (Thick v/s Thin Provisioning), and Network; Cloud Storage Providers: Amazon S3 and Google Big table.

Platform and software as a service (PaaS and SaaS):

PaaS: Definition and Characteristics, Examples: Google App Engine, Microsoft Azure;
SaaS: Definition and Characteristics, Service Oriented Architecture (SOA) and Web Services, Examples: Google Apps, Amazon Web Services.

Cloud service monitoring and management:

Service Level: Indicator (SLI), Objective (SLO), and Agreement (SLA), Service Availability and Reliability, SLA Calculation; Billing & Accounting; Comparing Scaling Hardware: Traditional vs. Cloud; Data Backup and Recovery, Performance Monitoring; Cloud Federation: Open Stack Inter-Cloud Resource Federation, Cloud Bursting.

Cloud security:

Infrastructure Security: Network, Host, and Application Level Security; Data Security and Storage: Data Privacy and Security Issues, Jurisdictional Issues raised by Data Location; Identity & Access Management: Access Control, Trust, Reputation, Risk; Authentication in Cloud Computing: Client Access, Commercial and Business Considerations.

Cloud applications:

Scientific Applications, High Performance Computing (HPC), Map Reduce, Online Video Streaming, Massively Multiplayer Online Game Hosting, Content Delivery Networks.

Open stack architecture:

Open Stack Architecture Overview, Services: Compute, Network, Storage (Image, Object, and Block), Identity, Dashboard, Orchestration, Metering & Monitoring.

Google cloud platform (GCP):

Google Cloud Security and Compliance, Data Encryption (at Rest, in Transit), Data Deletion Pipeline, European Union Data Protection Compliance.

REFERNECES

1. RajkumarBuyya, 'Mastering Cloud Computing: Foundations and Applications Programming', Morgan Kaufmann, 2013.
2. Barrie Sosinsky, 'Cloud Computing Bible', Wiley, 2011.
3. OpenStack Team, 'Red Hat OpenStack Platform 12 Architecture Guide', Red Hat, 2019.

RECOMMENDED READING

1. George Reese, 'Cloud Application Architectures', O'Reilly, 2009.
2. Nick Antonopoulos, 'Cloud Computing Principles, Systems and Applications', Springer, 2017.
3. Ronald L., 'Cloud Security', Wiley, 2010.

| | | |
|-----------------------|--------------------------------------|--------------------|
| Programme Name | S.Y.M.C.A. | SEMESTER II |
| Course Code | R5MC5114S | |
| Course Title | DATA WAREHOUSING (Elective 1) | |

PREREQUISITE

Knowledge of Data bases, spreadsheets is required for this course, and programming language like Java, R or Python is desired for this course.

COURSE OUTCOMES

1. Understand the various steps and architecture of DW / BI lifecycle.
2. Apply the concepts of ETL on a given data set.
3. Create various dimensional models for a data warehouse.
4. Design a data warehouse to present information collected to support decision making using OLAP and how different types of changes in warehouse are handled.
5. Distinguish between various space management techniques for DW.

COURSE CONTENTS

Introduction to data warehousing

Evolution of Data Warehousing (DWH), DWH concepts, benefits and problems, Goals of a Data Warehouse, Comparison of Databases and Data Warehousing, Additional Considerations, Data Realities, design issues, Common Pitfalls to avoid.

Data warehouse Architecture and its components

Data warehouse development life cycle, Data warehouse three tier architecture, Components: Source Data Component, Data Staging, Data Storage, Information Delivery, Data Warehouse Metadata, Management and control component, Reasons for creating a data mart and data mart issues, Importance of Metadata in DW.

Data Preprocessing and Querying:

Needs Preprocessing the data, Data cleaning, Data integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation., Handling missing data-various methods of replacing missing values, Identifying misclassifications Outlier introduction, Methods for identifying outliers. CUBE, ROLL UP and STAR queries.

Dimensional Modeling

Data Warehouse Design - DE normalization, STAR schema design, Snowflake schema, Fact Constellation Schema, Conforming Dimensions, Fundamental Concepts, Basic Fact Table Techniques, Basic Dimension Table Techniques, Integration via Conformed Dimensions , Dealing with Slowly Changing Dimension Attributes, Dealing with Dimension Hierarchies, Advanced Fact Table Techniques, Advanced Dimension Techniques.

ETL

ETL Process Overview, Develop the ETL Plan, Operational Source Systems, Data Extraction Source Identification, Data Extraction Techniques ,Evaluation of the Techniques Transformation: Cleaning and Conforming the data, and other appropriate transformations, Data Transformation Techniques – Minmax normalization, Z-score standardization, Data Loading , Applying Data: Techniques and Processes ,Data Refresh Versus Update , Procedure for Dimension Tables, Fact Tables: History and Incremental Loads , Incremental ETL Processing development, Delivering: Prepare for Presentation, Managing the ETL Environment, Real-Time Implications.

Online Analytical Processing (OLAP)

Introduction, Applications and Benefits, OLAP operations – need and types, OLAP tools – Categories and Codd's rules, Multidimensional OLAP, Relational OLAP, Hybrid OLAP, and Desktop OLAP.

Space Management in Data warehouse

Schemas for storing data in warehouse using different storage structures, B-tree index, hashindex, clusters, Bitmap index functionalindex, domain index, Data partitions.

RECOMMENDED READING

1. Kimball, Ralph. The data warehouse lifecycle toolkit. John Wiley & Sons, 2008.
2. Kimball Dimensional Modeling Techniques, John Wiley & Sons, 2008.
3. Ponniah, Paulraj. Data warehousing fundamentals: a comprehensive guide for IT professionals. John Wiley & Sons, 2004.
4. Berry, M., and G. Linoff. "Mastering data mining. 2000." JohnWiley& Sons, New York.
5. Inmon, William H. Building the data warehouse. John Wiley & sons, 2005.
6. ReemaTheraja , Data Warehousing, Oxford Higher Education press

| | | |
|-----------------------|---------------------------------------|--------------------|
| Programme Name | M.C.A | SEMESTER-II |
| Course Code | R5MC5121T | |
| Course Title | MULTIMEDIA SYSTEM (Elective 2) | |

PREREQUISITES

Good knowledge of Computer Graphics

COURSE OUTCOMES

1. Describe various multimedia elements
2. Distinguish between various data compression techniques used in multimedia applications.
3. Create a multimedia application.
4. Paraphrase data and file format standards specific to multimedia data.
5. Develop multimedia application utilizing animation models.

COURSE CONTENTS

Introduction, Media and Data Streams

Multimedia Elements; Multimedia Applications; Multimedia Systems Architecture; Evolving Technologies for Multimedia Systems; Defining Objects for Multimedia Systems; Multimedia Data Interface Standards; Data and File Format Standards; Classification of Multimedia; Multimedia and Hypermedia, World Wide Web, Overview of Multimedia Software Tools, Further Exploration, Multimedia Authoring, Some Useful Editing and Authoring Tools, VRML; Characterizing Continuous Media Data Streams.

Graphics and Images, Audio Video Technology, Computer-Based Animation

Capturing Graphics and Images, Computer Assisted Graphics and Image Processing; Reconstructing Images; Graphics and Image Output Options. Sound: Frequency, Amplitude, Sound Perception and Psychoacoustics; Audio Representation on Computers; Three Dimensional Sound Projection; Music and MIDI Standards; Speech Signals; Speech Output; Speech Input; Speech Transmission. Basics Television Systems; Digitalization of Video Signals; Digital Television; Basic Concepts: Specification of Animations; Methods of Controlling Animation; Display of Animation; Transmission of Animation.

Data Compression – 1

The need for Data Compression; Storage Space; Coding Requirements; Source, Entropy, and Hybrid Coding; Basic Compression Techniques; JPEG: Image Preparation, Lossy Sequential DCT-based Mode, Expanded Lossy DCT-based Mode, Lossless Mode, Hierarchical Mode

Data Compression – 2

H.261 (Px64) and H.263: Image Preparation, Coding Algorithms, DataStream, H.263+ and H.263L; MPEG: Video Encoding, Audio Coding, DataStream, MPEG-2, MPEG-4, MPEG-7; Fractal Compression.

Content Analysis

Simple Vs. Complex Features; Analysis of Individual Images; Analysis of Image Sequences; Audio Analysis; Applications.

Multimedia Communication

Fundamentals of data communication and networking, Bandwidth requirements of different media, Real time constraints: latency, video data rate, multimedia over LAN and WAN, Multimedia conferencing, video-on-demand broadcasting issues.

Multimedia Information Systems

Operating system support for continuous media applications: Media stream protocol, file system support for continuous media, data models for multimedia and hypermedia information, multimedia servers, databases and content management.

Multimedia Application Design

Multimedia Application Classes; Types of Multimedia Systems; Virtual Reality Design and Modeling Language; Components of Multimedia Systems; Organizing Multimedia Databases; Application Workflow Design Issues; Distributed Application Design Issues.

REFERENCES

1. Prabhat K. Andleigh, KiranThakrar, Multimedia Systems Design PHI, 2003
2. Ralf Steinmetz, KlaraNarstedt, Multimedia Fundamentals: Vol 1-Media Coding and Content Processing Pearson Education / PHI, 2ndEdition, 2003.

RECOMMENDED READING

1. K.R.Rao, Zoran S., Bojkovic and Dragorad A. Milvanovic, Multimedia Communication Systems: Techniques, Standards, and Networks , Pearson Education, 2002.
2. Nalin K Sharad, Multimedia Information Networking PHI, 2002.
3. Jerry D. Gibson, Toby Berger, Tom Lookabaugh, Dave Lindergh and Richard L. Baker Digital Compression for Multimedia: Principles and Standards Elsevier, 2006
4. Ralf Steinmetz and KlaraNahrstedt, Multimedia: Computing, Communications, and Application, Prentice Hall, 1995.

| | | |
|-----------------------|---|--------------------|
| Programme Name | M.C.A | SEMESTER-II |
| Course Code | R5MC5121T | |
| Course Title | MULTIMEDIA SYSTEM LAB (Elective 2) | |

PREREQUISITES

1. Knowledge of programming language
2. Good knowledge of Computer Graphics

COURSE OUTCOMES

1. Develop multimedia applications using various multimedia tools.
2. Implement different compression algorithm.
3. Develop interactive animations using multimedia tools
4. File format conversion.

TITLE OF EXPERIMENT

1. Study of multimedia I/O devices
2. Perform animation using any Animation software
3. Creating animated e-card using macromedia Flash
4. Study and implement the concept of discrete cosine transform (DCT)
5. Study and implement the concept of LampelZiv Welch (LZW) Coding
6. Study and implement arithmetic coding to encode a sequence of symbols
7. Create multimedia database for student ID card preparation
8. Study and implement the concept of differential pulse code modulation
9. Record a speech and perform compression and decompression
10. Design a game/application in flash.
11. Convert BMP file to JPG file using any programming language.

ASSESSMENT

The distribution of marks for ESE practical exam are as follows:

- **Viva-Voice : 20 Marks**
- **Practical test /Quiz : 20 Marks**

| | | |
|-----------------------|-------------------------------------|--------------------|
| Programme Name | F.Y. M.C.A. | SEMESTER II |
| Course Code | R5MC5122T | |
| Course Title | IMAGE PROCESSING(Elective 2) | |

PREREQUISITE

A good knowledge of applied mathematics

COURSE OUTCOMES

1. Explain the basic elements and applications of image processing.
2. Apply various transformation techniques for Image enhancement.
3. Compare and contrast image compression techniques.
4. Discuss various morphological techniques.
5. Design and develop image processing applications in practice.

COURSE CONTENTS

Introduction

Introduction to Digital Image processing, Application, steps in image processing, various components, digital image fundamentals, Image sampling and quantization, relationship between pixels.

Intensity transformations

Some basic intensity transforms: Introduction to Unitary Transform, Discrete Fourier Transform (DFT), Properties of DFT, Fast Fourier Transform (FFT), Discrete Hadamard Transform (DHT), Fast Hadamard Transform (FHT), Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT), Discrete Fourier transform, Basic of filtering in frequency domain, restoration of noisy image

Image enhancement:

Histogram Processing, Neighbourhood Processing, Fundamentals of spatial filtering, smoothing, sharpening, Homomorphic Filtering, combining spatial enhancement methods,

Color image processing

Color fundamentals, color models, color transformation, smoothing and sharpening, image segmentation based on color, noise in color image, color image compression, image compression techniques: Lossless Compression: Run Length Coding, Arithmetic Coding, Huffman Coding, Differential PCM, Lossy Compression: Improved Gray Scale Quantization, Vector Quantization, JPEG, MPEG-1.

Image restoration and reconstruction

Noise models, restoration in presence of noise, periodic noise reduction

Morphological image processing

Binary Morphological Operators, Hit-or-Miss Transformation, Boundary Extraction, Region Filling, Erosion and dilation Thinning and Thickening, Connected Component Labeling, Iterative Algorithm and Classical Algorithm Gray scale morphology, Image segmentation

Applications of image processing

Character Recognition, Digital Watermarking, Finger, Iris, Face Recognition, Vehicle Number Plate Recognition, Industrial Applications, CBIR.

REFERENCES

1. Gonzales and Wood, Digital Image processing ,Pearson 3rd Edition
2. Jain, Anil K. Fundamentals of digital image processing. Prentice-Hall, Inc.,1989.

RECOMMENDED READING

1. Chanda, Bhabatosh, and Dwijesh Dutta Majumder. Digital image processing and analysis. PHI Learning Pvt. Ltd.,2004.
2. Dr. Bernd Jahne, Digital Image processing, SpringerIndia

| | | |
|-----------------------|--|--------------------|
| Programme Name | F.Y. M.C.A. | SEMESTER II |
| Course Code | R5MC5122P | |
| Course Title | IMAGE PROCESSING LAB (Elective 2) | |

PREREQUISITE

Knowledge of C / C++ / Matlab / Python / R is required for this course

COURSE OUTCOMES

1. Implement image sampling and quantization
2. Apply histogram equalization for image enhancement
3. Use morphological operations on image
4. Implement and compare image compression techniques
5. Design and develop image processing applications in practice.

TITLE OF EXPERIMENT

1. Write a Program to study the effects of reducing the quantization values and spatial resolution.
2. Write a Program to study the effects intensity transformation.
3. Write a program to study the effects of image enhancement.
4. Write a program to plot histogram and apply histogram equalization.
5. Write a program to apply masking effect on the image.
6. Write a program to apply morphological operation on image
7. Write a program to study the effect of lossy and lossless Image Compression.
8. Write a program to perform various edge detection techniques on an Image.
9. Write a program to perform the content-based image recognition (CBIR) technique.

ASSESSMENT

The distribution of marks for ESE practical exam is as follows:

- **Viva-Voice : 20 Marks**
- **Practical test /Quiz : 20 Marks**

| | | |
|-----------------------|---|--------------------|
| Programme Name | M.C.A | SEMESTER-II |
| Course Code | R5MC5123T | |
| Course Title | SOFTWARE DESIGN PATTERN (ELECTIVE 2) | |

PREREQUISITES

1. Object Oriented Analysis and Design.
2. Data structures and algorithms.
3. Programming Language (C++ or Java)

COURSE OUTCOMES

1. Demonstrate of patterns related to object oriented design.
2. Describe the design patterns that are common in software applications.
3. Analyze a software development problem and express it.
4. Design a module structure to solve a problem, and evaluate alternatives.
5. Implement a module so that it executes efficiently and correctly.

COURSE CONTENTS

Introduction to Design Pattern:

What is a Design Pattern?, Design Patterns in Smalltalk MVC, Describing Design Patterns, The Catalogue of Design Patterns, Organizing The Catalog, How Design Patterns solve Design Problems, How to Select a Design pattern, How to Use a Design Pattern.

A Case Study: Designing a Document Editor:

Design Problems, Document Structure, Formatting, Enabling the user interface, Supporting Multiple Look-and-Feel Standards, Multiple Window Systems, User operations, Spelling Checking and Hyphenation

Creational Patterns

Abstract Factory, Builder, Factory Method, Prototype, Singleton

Structural Patterns

Adapter, Bridge, Composite, Decorator, Façade, Flyweight, Proxy

Behavioral Patterns

Chain of Responsibility, Command, Interpreter, Iterator, Mediator, Memento, Observer, State Strategy, Template Method, Visitor

Conclusion

What to Expect from Design Patterns, Brief History, Pattern Community, Invitation, and Parting Thought

REFERENCE

1. Erich Gamma, Design Patterns, Pearson Education

RECOMMENDED READING

1. Mark Grand, Patterns in JAVA Vol-I (or) Vol-II, Wiley Dream Tech.
2. Mark Grand, Java Enterprise Design Patterns Vol-III, Wiley Dream Tech.

| | | |
|-----------------------|---|--------------------|
| Programme Name | M.C.A | SEMESTER-II |
| Course Code | R5MC5123P | |
| Course Title | SOFTWARE DESIGN PATTERN LAB (ELECTIVE 2) | |

PREREQUISITES

1. Object Oriented Analysis and Design.
2. Data structures and algorithms.
3. Programming Language (C++ or Java)

COURSE OUTCOMES

1. Identify the purpose and methods of common object-oriented design patterns.
2. Develop programming for basic object oriented programming concepts.
3. Select and apply these patterns in their own designs for simple programs.
4. Create documentation outlining the testable and complete design of a simple program.
5. Produce and present documents for the purpose of capturing software requirements and specification.

TITLE OF EXPERIMENT

1. Introduction to Design patterns. Describe the organization of catalog along with the following design patterns.
 - A) Creational Patterns.
 - B) Structural Patterns.
 - C) Behavioral Patterns.
2. Implement the following concepts in java.
 - A). Method overriding.
 - B). Interface.
 - C). Abstract class.
3. A) Design a Program to implement Factory pattern.
B) Design a program to implement abstract factory
4. A) Design a Program to implement Singleton pattern.
B) Design a Program to implement Composite design pattern.
5. A) Design a program to implement decorator pattern.
B)Design a Program to implement proxy design pattern.
6. A) Design a Program to design chain of responsibility pattern.
B) Design a Program to design mediator pattern.
7. Design a program to implement iterator pattern.
8. Design a program to implement visitor pattern.
9. Case Study: Railway Reservation System
10. Describe the Methods to analyze the complexity of design patterns.

ASSESSMENT

The distribution of marks for ESE practical exam is as follows:

- **Viva-Voice : 20 Marks**
- **Practical test /Quiz : 20 Marks**

| | | |
|-----------------------|-------------------------------------|--------------------|
| Programme Name | F.Y. M.C.A. | SEMESTER II |
| Course Code | R5MC5124T | |
| Course Title | ETHICAL HACKING (Elective 2) | |

PREREQUISITES

1. Basic concepts of the operating system, networking, and database
2. Knowledge of programming language
3. Basic knowledge of information security

COURSE OUTCOMES

1. Distinguish between hacking and ethical hacking.
2. Describe basic concepts of computer systems and information security.
3. Elaborate on basic hacking methods.
4. Analyze hacking of web applications, wireless networks, and mobile platforms.
5. Evaluate various security assessment and defense mechanisms against hacking.

COURSE CONTENTS

Introduction to ethical hacking:

Overview of Ethics and Ethical Hacking; Types of Hacking and Hackers; Advantages and Disadvantages of Hacking; Purpose of Hacking; Methodology of Ethical Hacking: Reconnaissance and Foot printing, Scanning and Enumeration, Gaining Access, Maintaining Access, Covering Tracks.

Understanding of computer systems and information security

Operating System Basics; Database Basics; Networking Basics; Programming Basics for Security: Windows PowerShell, Linux Shell Scripting, Python; Virtualization and Cloud Basics; Information Security Basics: CIA, AAA, Vulnerability, Risk, Threat, and Exploit, Security Policies.

Hacking basics:

Password-Cracking Techniques, Key loggers, Trojans, Viruses, Computer Worms, Rootkits, Online Malware Analysis, Social Engineering, Privilege Escalation, Denial of Service Attack, Botnet, Alternate Data Streams, Steganography, Covering Tracks, Reconnaissance: Foot printing, Scanning Networks, Enumeration.

Hacking web applications, wireless networks, and mobile platforms

Web Application Hacking: Attack Vectors, Flaws, Hacking Web Servers; Wireless Hacking: SSID, Wi-Fi Authentication, Searching for Wi-Fi Networks, Spectrum Analysis, Bluetooth Hacking; Hacking Mobile Platforms: Mobile Terminology, Attack Vectors, Android OS, iOS, Mobile Device Management.

Security assessment and defense mechanisms:

Penetration Testing: Why Security Assessments Are Required, Deciding What Should Be Tested, The Penetration Testing Lifecycle; IDSes, Firewalls, and Honeypots; Cryptography: Objectives, Types, Tools, Message Digests, PKI, SSL; Security Architecture and Design: Data Classification, Security Models, Application Architecture, and Security Architecture.

REFERENCES

1. SagarRahalkar, 'Certified Ethical Hacker (CEH) Foundation Guide', Apress, 2016.
2. Ric Messier, 'CEH v10 Certified Ethical Hacker Study Guide', Sybex, 2019.
3. SagarRahalkar, 'Quick Start Guide to Penetration Testing', Apress, 2019.

RECOMMENDED READING

1. Matt Walker, 'CEH All-in-One Exam Guide', McGraw-Hill 4ed, 2019.
2. Patrick Enebreton, 'The Basics of Hacking and Penetration Testing', Syngress 2ed, 2013.
3. Georgia Weidman, 'Penetration Testing - A hands-on introduction to Hacking', No Starch Press, 2014.

| | | |
|-----------------------|----------------------------|--------------------|
| Programme Name | F.Y. M.C.A. | SEMESTER II |
| Course Code | R5MC5124P | |
| Course Title | ETHICAL HACKING LAB | |

PREREQUISITES

1. Basic knowledge of programming language and networking.
2. Basic knowledge of information security and Linux operating system.

COURSE OUTCOME

1. Perform system and network information gathering.
2. Perform various attacks on the system and network using various tools.
3. Identify different hacking techniques and security measures.

TITLE OF EXPERIMENT

1. Execute and analyze Linux commands ifconfig, ping, traceroute, nslookup, netstat; analyze file access & permissions, and user-groups & admin privileges; listen to network traffic using Wireshark.
2. Find the registrant details of any website-domain using WHOIS lookup; test the SSL certificate of the website using SSLLABS.com; and explore NETCRAFT.com for website foot printing.
3. Perform SQL-Injection attack on a website using SQLMAP tool.
4. Using NMAP tool, detect the OS on a target-system, perform UDP & intense scan, and use various NMAP scripts for enumerating services on a target system.
5. Use OpenVAS to scan one Windows host and one Unix-based host and generate vulnerability reports in HTML and PDF.
6. Browse through the directory of the Metasploit framework and understand its structure. Try various commands such as set, setg, unset, unsetg, spool. Explore various auxiliary modules and use them to scan services such as HTTP, FTP, SSH, and so on. Try different features of Meterpreter such as getsystem and hashdump.
7. Download and install ZenMAP tool and explore various options for network scanning & enumeration.
8. Explore 'SNORT', a free open source network intrusion detection and prevention system.

ASSESSMENT

The distribution of marks for ESE practical exam is as follows:

- **Viva-Voice : 20 Marks**
- **Practical test /Quiz : 20 Marks**

| | | |
|-----------------------|---------------------------------------|--------------------|
| Programme Name | M.C.A | SEMESTER-II |
| Course Code | R5MC5125T | |
| Course Title | INTERNET OF THINGS(Elective 2) | |

PREREQUISITES

Knowledge of networking and various mobile communication protocols.

COURSE OUTCOMES

1. Analyze the failure of TCP in WSN and study/learn different types of protocol.
2. Apply the knowledge of MANET for drawing information and translating it to actuator.
3. Describe the different types of sensors and Actuators.
4. Apply information centric sensing for configuring the Network topology.

COURSE CONTENTS

Internet of things (IOT) overview

What Is the Internet of Things (IOT), Background and More Complete IOT Definition, How to Monitor and Control Things from Anywhere in the World? , Why Do We Want to Monitor and Control Things? , Who Will Monitor and Control? How Is Security Guaranteed? , IOT Reference Framework, Why Now? The 12 Factors for a Perfect Storm, Convergence of IT and OT, the Astonishing Introduction of Creative Internet-Based Businesses, Mobile Device Explosion.

The Internet in IOT

The Open System Interconnection Model, End-to-End View of the OSI Model, Transmission Control Protocol/Internet Protocol (TCP/ IP), TCP/IP Layer 4: Application Layer, TCP/IP Layer 3: Transport Layer, TCP/IP Layer 2: Internet Layer, TCP/IP Layer 1: Network Access Layer.

The things in IOT: sensors and actuators

IOT Sensors, RFID, RFID Main Usage and Applications, Video Tracking, Video Tracking Applications, IOT Actuators, Actuator Types, Controlling IOT Devices, How Things Are Identified in IOT?

IOT protocol stack: a layered view

Link Layer, Internet Layer, Application Protocols Layer, Data Serialization Formats , Communication Paradigms, Survey of IOT Application Protocols, Application Services Layer.

IOT services platform: functions and requirements

IOT Services Platform Functions, IOT Platform Manager, Communication Manager, Data Management and Repository ,Element Manager, Configuration Management ,Fault Management, Performance Management, Important Performance Measures for IOT Devices (e.g., Sensors), Security Management , Firmware Manager, Topology Manager , Group Manager, Billing and Accounting , Subscription and Notification Manager ,API Manager

Internet of things security and privacy

IOT Security Challenges, IOT Security Requirements, IOT Three-Domain Architecture, Cloud Domain Attacks and Counter measures, Fog Domain Attacks and Countermeasures, Sensing Domain Attacks and Countermeasures, Summary and Future Directions

IOT vertical markets and connected ecosystems

IOT Verticals, IOT Agriculture and Farming, IOT Energy Solutions, IOT Oil and Gas Solutions, IOT Smart Building Solutions, IOT Finance, IOT Healthcare, IOT Industrial, IOT Retail, IOT Transportation, IOT Service Model: Anything as a Service , Thrust as a Service, Imaging as a Service , Farming as a Service, IT as a Service , Enabling “Anything as a Service” , Connected Ecosystems, IOT Services Terminologies, IOT Connected Ecosystems Models , IOT Connected Ecosystems Models Key Capabilities.

The role of open source in IOT

The Open Source Movement, Why Open Source? , Drivers for Open Source Consumers, Drivers for Open Source Contributors, Open Source vs. Standards, Open Source Partnering with Standards, A Tour of Open Source Activities in IOT, IOT Devices, IOT Services Platform

TEXTBOOK:

1. Internet of Things from Hype to Reality, Second Edition, by Ammar Rayes • Samer Salam. Springer Nature Switzerland AG 2019 A. Rayes S. Salam, Internet of Things From Hype to Reality,
2. M. Dabbagh, B. Hamdaoui, M. Guizani and A. Rayes, Release-time aware VM placement. in Globecom Workshops (GC Wkshps), (2014), pp. 122–126

REFERENCES:

1. M. Dabbagh, B. Hamdaoui, M. Guizani, A. Rayes, toward energy-efficient cloud computing: Prediction, consolidation, and over commitment. Network, IEEE 29(2), 56–61 (2015)
2. M. Dabbagh, B. Hamdaoui, M. Guizani, A. Rayes, Efficient datacenter resource utilization through cloud resource over commitment, in IEEE Conference on Computer Communications Workshops (INFOCOM WKSHS), 2015, pp. 330–335

| | | |
|-----------------------|--|--------------------|
| Programme Name | M.C.A | SEMESTER-II |
| Course Code | R5MC5125P | |
| Course Title | INTERNET OF THINGS LAB (Elective 2) | |

PREREQUISITES

Basic knowledge of electronic, C, C++, python.

COURSE OUTCOMES

1. Use of components of Arduino/Raspberry Pi and relevant software.
2. Interface the different components of IOT device.
3. Study the MQTT broker, TCP and UDP server in IOT.

TITLE OF EXPERIMENT

1. In Internet of Things (IOT) Familiarization with Arduino/Raspberry Pi and perform necessary software installation.
2. Write a program to interface LED/Buzzer with Arduino/Raspberry Pi and write a program to turn ON LED for 1 sec after every 2 seconds.
3. Write a program to interface Push button/Digital sensor (IR/LDR) with Arduino/Raspberry Pi and write a program to turn ON LED when push button is pressed or at sensor detection.
4. Write a program to interface DHT11 sensor with Arduino/Raspberry Pi and write a program to print temperature and humidity readings
OR
Write a program to interface OLED with Arduino/Raspberry Pi and write a program to print temperature and humidity readings on it.
5. Write a program to interface motor using relay with Arduino/Raspberry Pi and write a program to turn ON motor when push button is pressed.
6. Write a program to interface Bluetooth with Arduino/Raspberry Pi and write a program to send sensor data to smartphone using Bluetooth.
7. Write a program on Arduino/Raspberry Pi to upload temperature and humidity data to thing speak cloud.
8. Write the experiment to install MySQL database on Raspberry Pi and perform basic SQL queries.
9. Write a program on Arduino/Raspberry Pi to publish temperature data to MQTT broker and subscribe to MQTT broker for temperature data and print it.
10. Write a program to create TCP/UDP server on Arduino/Raspberry Pi and respond with humidity data to TCP/UDP client when requested.

ASSESSMENT

The distribution of marks for ESE practical exam is as follows:

1. Viva-Voice : 20 Marks
2. Practical test /Quiz : 20 Marks

| | | |
|-----------------------|-----------------------------------|--------------------|
| Programme Name | M.C.A | SEMESTER-II |
| Course Code | R5MC5027L | |
| Course Title | PROFESSIONAL COMMUNICATION | |

PREREQUISITE

Ability to communicate in correct English

COURSE OUTCOMES

1. Apply the principles and practices of business communication for communicating in a professional environment.
2. Deliver formal presentations employing effective range of verbal and non verbal skills
3. Design a technical document with correctness of language, appropriate vocabulary and style.
4. Recognize the attributes of a suitable candidate for a job, through participation in group discussion, interview and resume writing.
5. Demonstrate knowledge of professional ethics and behavior

COURSE CONTENTS

Basics of Business Communication

- a. Concept and meaning of communication
- b. Types of communication
- c. Verbal and non-verbal communication
- d. barriers to the process of communication
- e. Channels of communication
- f. Role of communication in in formationage

Speaking

- a. Intonation
- b. Modulation
- c. Basics of publicspeaking
- d. Gaining confidence

Presentation Skills

- a. Oral presentation
- b. Graphic presentation

Career Oriented Communication

- a. Resume, Language and format of job application
 - b. Job Interviews
 - i. Purpose and process
 - ii. How to prepare for interviews
 - iii. Language and style to be used in interview
 - iv. Types of interview questions and how to answer them
- Group Discussion: structure, dynamics and techniques of effective participation

Technical Writing

- a. Technical writing process
 - b. Style and organization in technical writing
 - c. objectivity, clarity, precision as defining features of technical communication
- Language and format of various types of business letters, reports; proposals, e-mails, minutes of meeting, research paper

Introduction to Corporate Ethics

and etiquettes Inter personal

Communication

- a. Working and communicating in teams
- b. Assertive behavior
- c. Negotiation skills
- d. Time management.

Text Books:

1. Hory Shankar Mukharjee, Business Communication, OUP
2. McGRATH, E. H. S. J. Basic managerial skills for all. PHI Learning Pvt.Ltd., 2011.

Recommended Reading

3. Rizvi, M. Ashraf. Effective technical communication. TataMcGraw-Hill, 2005.
4. Meenakshi Raman, Business Communication Prakash Singh, OUPR. Subramanian, Professional Ethics OUP

| | | |
|-----------------------|----------------------------|--------------------|
| Programme Name | M.C.A | SEMESTER-II |
| Course Code | R5MC5025L | |
| Course Title | JAVA AND PYTHON LAB | |

PREREQUISITES

1. Knowledge of C/C++, Object Oriented Concepts and good programming skills

COURSE OUTCOMES

1. Develop applications in Java using concepts of OOP, Interfaces and Packages.
2. Create an interactive GUI application in Java having database connectivity.
3. Design and develop an applet based application.
4. Implement Java Programs using built in java classes.
5. Develop Python application using object oriented concepts

TITLE OF EXPERIMENT

JAVA

1. Refactor following code:

```

import java.util.Date;
import java.util.HashMap;
import java.util.Map;
import java.util.UUID;

public class NotificationService {
    private final MapNotificationDAO storage = new MapNotificationDAO();

    public UUID raiseNotification() {
        return this.storage.addNotification(new Date());
    }

    public Date getNotificationTime(UUID id) {
        return this.storage.getNotification(id);
    }

    public static void main(String[] args) {

    }

}

class MapNotificationDAO {
    private final Map<UUID, Date> notifications = new HashMap<UUID, Date>();

    public UUID addNotification(Date time) {

```

```

        UUID id = UUID.randomUUID();
        this.notifications.put(id, time);
        return id;
    }

    public Date getNotification(UUID id) {
        return this.notifications.get(id);
    }

```

- a. Create a new package-private class NotificationVO which would have notification date (data type : java.util.Date) and notification message (data type : java.lang.String)
 - b. Create a new *package-private* interface, named *NotificationDAO*, that contains the same methods as *MapNotificationDAO*.
 - c. Refactor the *NotificationService* and *MapNotificationDAO* classes:
 - i. Modify NotificationDAO interface method addNotification to accept additional parameter for notification message (data type : String) and change getNotification to return NotificationVO
 - ii. Modify NotificationService interface to:
 1. Change method raiseNotification to accept parameter for notification message (data type : String)
 2. Change getNotificationTime implementation based on changed signature of NotificationDAO::getNotification method
 3. Add method getNotificationMessage which would accept parameter of type UUID and return notification message
 - iii. *MapNotificationDAO* should implement the *NotificationDAO* interface and implement modified addNotification and getNotification methods
 - iv. *NotificationService* should have a constructor that accepts *NotificationDAO*.
 - v. The *raiseAlert* and *getAlertTime* methods should use the object passed through the constructor.
 - d. Write a code inside main() method to accept count of alerts as 1st argument and subsequently read new line delimited notification messages from command line and add those notifications and print those notifications
- 2.

```

public class UserInputBuilder {

    public static class TextInputBuilder {
        public void add(char c) {
            //Write a code here
        }
    }
}

```

```

    }

    public String getValue() {
        //Write a code here
    }
}

public static class NumericInputBuilder extends TextInputBuilder {
    @Override
    public void add(char c) {
        //Write a code here
    }
}

public static void main(String[] args) {
    TextInputBuilder input = new NumericInputBuilder();
    input.add('1');
    input.add('a');
    input.add('0');
    System.out.println(input.getValue());
}
}

```

UI widget factory needs to be designed. There are two types of input builders: *TextInputBuilder*, which accepts all characters and *NumericInputBuilder*, which accepts only digits.

- a. Implement the class *TextInputBuilder* that contains:
 - Public method *void add(char c)* - adds the given character to the current value
 - Public method *String getValue()* - returns the current value
- b. Implement the class *NumericInputBuilder* that:
 - Inherits from *TextInputBuilder*
 - Overrides the *add* method so that each non-numeric character is ignored
- c. Implement the class *OddNumericInputBuilder* that
 - Inherits from *NumericInputBuilder*
 - Overrides the *add* method so that only odd numeric character is accepted
- d. Implement the class *EvenNumericInputBuilder* that
 - Inherits from *NumericInputBuilder*
 - Overrides the *add* method so that only even numeric character is accepted

- e. Write a main method in `UserInputBuilder` class to create and test each builder implementation class
3. Write a program to
 - a. Read two arguments from command line :
 - i. File directory path
 - ii. Search text
 - b. Recursively traverse all directories and files and print name of file and count of search text within that file

(Note:- Program should use Java concurrency API (`java.util.concurrent.ExecutorService`) for parallel processing of file search)
4. Design the program to demonstrate the use of Lambda expressions, Optional, Streams operations (Map,Reduce,Filter)
5. Design the program to demonstrate difference in `LocalDate`, `LocalDateTime`. Create a function which would return difference in days between two dates.

PYTHON

1. Implement a `groupAndSortOwners` function that:
 Accepts a dictionary containing the file owner name for each file name.
 Returns a dictionary containing a list of file names for each owner name (i.e. grouped by owner name), in alphabetical order of file names.
 For example, for dictionary `{'Input.txt': 'Albert', 'Code.py': 'Stanley', 'Output.txt': 'Albert', 'btech.txt': 'Albert'}` the `groupAndSortOwners` function should return `{'Albert': ['btech.txt', 'Input.txt', 'Output.txt'], 'Stan': ['Code.py']}`.
2. Write a program to print following pattern (Note:- No of rows of pattern to be accepted as input for e.g. given below is pattern for 4 rows):


```

31  29  27  25  23  21  19
    17  15  13  11  9
      7   5   3
        1
      
```
3. Write a class `QuadraticEquationSolver` which would have method `findRoot` to identify roots of quadratic equation. Quadratic equations take a form of $ax^2 + bx + c = 0$
 Formula to identify roots of quadratic equation is

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

For e.g. $2x^2 - 5x - 3 = 0$ has roots as **12** and **-2**.

findRoots method would accept coefficients of quadratic equation (i.e. values of a, b & c) and return both roots.

4. Design a program to implement classes, Inheritance, overloading and overriding
5. Design a program for exception handling, multithreading and database operations

ASSESSMENT

The distribution of marks for ESE practical exam are as follows:

- **Viva-Voice : 20 Marks**
- **Practical test /Quiz : 20 Marks**