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 2019)

For

First Year

Of

Three Year Postgraduate Program Leading to

Master of Computer Applications (MCA)

Implemented from the batch admitted in First Year, 2019-20

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 f 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, transparence 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 utilized for efficient communication.

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.	R4MC5011S	Discrete Mathematics	3	1	-	4.0	20	20	60	3
2.	R4MC5012T	C and C++ Programming	3	-	-	3.0	20	20	60	3
	R4MC5012P	C and C++ Programming Lab	-	-	4	2.0	100% CIE			
3.	R4MC5013T	System Modeling	3	-	-	3.0	20	20	60	3
	R4MC5013P	System Modeling Lab	-	-	2	1.0				
4.	R4MC5014S	Computer Organization and Architecture	3	-	-	3.0	20	20	60	3
5.	R4MC5015T	Data Communication and Network	3	-	-	3.0	20	20	60	3
	R4MC5015L	Data Communication and Network Lab	-	-	2	1.0	100% CIE			
6.	R4MC5016L	Web Technology Lab	-	-	2	1.0	100% CIE			
7.	R4MC5018L	*Business English	-	-	2	P/NP	100% CIE			
		Total	15	1	12	21				

*Pre-test will be given and those who pass the test, concession will be given in term of attendance, if they do not clear the pre-test then attendance is compulsory.

Abbreviations: **L**: Lecture, **T**: Tutorial, **P**: Practical, **TA**: Teacher Assessment / Term work Assessment, **MST**: In Semester Test, **ESE**: End Semester Written Examination, **CIE**: Continuous In-semester Evaluation

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.	R4MC5021S	Probability and Statistics	3	1	--	4.0	20	20	60	3
2.	R4MC5022T	Core and advanced Java Programming	3	-	-	3.0	20	20	60	3
	R4MC5022L	Core and advanced Java Programming Lab	-	-	3	1.5	100% CIE			
3.	R4MC5023S	Accounting Banking and Finance	3	-	-	3.0	20	20	60	3
4.	R4MC5024T	Data and file Structures	3	-	-	3.0	20	20	60	3
	R4MC5024L	Data and File Structures Lab	-	-	2	1.0	100 % CIE			-
5.	R4MC5025T	Operating Systems	3	-	-	3.0	20	20	60	3
	R4MC5025L	Operating Systems Lab	-	-	2	1.0	100% CIE			
6.	R4MC6015S	Professional and Technical Communication	1	-	2	2.0	20	20	60	3
7.	R4MC5028P	Unix Programming Lab	-	-	2	1.0	100% CIE			-
8.	R4MC5027D	Mini Project 1	-	-	3	1.5	100 % CIE			-
		Total	16	1	14	24				

Abbreviations: **L**: Lecture, **T**: Tutorial, **P**: Practical, **TA**: Teacher Assessment / Term work Assessment, **MST**: In Semester Test, **ESE**: End Semester Written Examination, **CIE**: Continuous In-semester Evaluation

Programme Name	*F.Y.M.C.A*	*Semester- I*
Course Code	**R4MC5011S**	
Course Title	**Discrete Mathematics**	

PREREQUISITE

A good knowledge of C, Data structure and Basic mathematics.

COURSE OUTCOMES

1. Design minimal Boolean function to achieve specified task using only one type of basic function using equivalence of formula.
2. Apply the concepts of sets, relations and functions to solve simple real life problems.
3. Apply graph theory model of data structure to solve problem of connectivity and constraint satisfaction and optimization.
4. Design codes to support varied real life problems.
5. Apply the knowledge of tree as a structure to build expression trees, minimum spanning tree and compression code to application problems

COURSECONTENTS

Propositions & logical operations

Notation, Connections, Normal forms, Truth tables, Equivalence & implications, Theory of inference for statement calculus, Predicate calculus, Rules of logic, Mathematical induction & Quantifiers.

Sets, Relations & Diagraphs

Review of set concepts, Relation & Diagraphs, Properties of relations Transitive closure Warshall Algorithm, Equivalence relations, Computer representation of relations and Diagraphs, Manipulation of relations, Partially ordered sets (posets), Hamming Codes

Order Relations and Structures

Partially ordered sets (posets), external elements of partially ordered sets. Lattices, finite Boolean algebra, Functions on Boolean algebra, Boolean functions as Boolean polynomials.

Tree

General tree, Binary tree, Tree traversals, Expression tree, building tree Using traversals, tree and sorting, weighted tree and prefix codes, Minimum Spanning Tree, Application of Trees (Huffman code).

Graph Theory

Definition, Path, circuits, reachability, connectedness, Matrix representation of graphs, List structures and graphs, PERT related techniques, Eularian and Hamiltonian graphs, Planar graphs, Euler paths and circuits, Hamiltonian paths and circuits, Graph coloring and Chromatic Polynomials, Applications of graph theory.

Groups and Applications

Monoids, semi groups, Product & quotients of algebraic structures, Isomorphism, homomorphism, automorphism, Normal subgroups, codes & groupcodes.

Optimization and Matching

Dijkstra‘s shortest path algorithm, Prim‘s and Kruskal‘s algorithms, Transport networks- Max flow and Min cut theorem, Matching theory.

Text Books

1. Kolman, Busby, Ross, Discrete Mathematical Structures Prentice Hall India, 5 th Edition, 2004.
2. C L Liu, Elements of Discrete Structures, McGraw Hill Education, 4th Edition,2012

Recommended Reading

3. Tremblay, Manohar, Discrete Mathematical Structures, Tata McGraw-Hill Education, 2001.
4. Rosen, Discrete Mathematics and Its Applications McGraw Hill Education, 7th Edition 2012.
5. Semyour Lipschutz, Varsha Patil, Discrete Mathematics, McGraw Hill Education, 3rd Edition, 2013.
6. Robert J. McElice, Robert Ash, Carol Ash, Introduction to Discrete Mathematics, McGraw Hill Education, 1st Edition, 1989.

Programme Name	*F.Y.M.C.A*	*Semester- I*
Course Code	**R4MC5012T**	
Course Title	**C++ Programming**	

PREREQUISITE

COURSE OUTCOMES

1. Apply knowledge of computer fundamental, specify design aspects to reduce complexity of a real world problem
2. Create, select and apply appropriate data structures to efficiently handle memory requirements of a particular model..
3. Demonstrate the knowledge and understanding of the object oriented concepts to solve environmental and societal issues..
4. Use research based knowledge to store and interpret the data with appropriate tools, to reduce the cost of the design, increasing the re-usability of the code

COURSE CONTENTS

Introduction to C

Why do we need programming?

Data Types in C

Data Types, Constant, Symbolic Constant, Scope of variable, Type casting, standard default values, C Literals

Operator and Expression

Arithmetic Operators, Bit wise Operator, Relational Operators, Boolean Logical Operators, Assignment Operator Increment and Decrement Operator, Conditional Operator, Special Operator, Operator Precedence

Decision making and Branching Statement

If statement, if else statement, the else if ladder, The switch statement, The? : Operator, While statement, do while statement, for statement, Nested Loops, Jumps in Loops, Labeled Loops, breaks, continue statements

.

Modular programming

Function components, passing data to the function, function return data type, passing by reference, macros, and inline function. Recursion, Storage classes

Arrays, Structure strings and Unions

Introduction to arrays, Operation on arrays, Array initialization, Two Dimensional Array, Operation on one and two dimensional arrays, Multidimensional Arrays, passing arrays to the function.

Structure, Union and Enumerated Data Types

Structures, declaration definition accessing members, nested structures, array of structures. Unions, passing structure to the functions.

Strings

Reading Strings, Writing Strings, string operations like concatenation, reversing, extracting, inserting, deleting, array of strings, passing of string to function

Pointers

Memory management: New and Delete, pointers to objects, Pointers to objects, this pointer, Pointer to functions

Introduction to C++

Why Do We Need Programming? Procedural Languages vs Object-Oriented Approach Characteristics Of Object-Oriented Language (Objects, Classes, Abstraction, Overloading, Inheritance, and Polymorphism Dynamic Binding, Resilience to change, Reusability of Code, Modularity of Code)

Classes, Object and Methods

Class Fundamentals, Declaring and Creating object, Accessing class, members and methods. Subclasses, scope of the function. Concept of constructor and destructors

Object initialization and Clean up

Constructor, parameterized Constructor, Constructor Overloading, Destructors order of construction and destruction, Static Member with constructors and destructors

Operator Overloading

Introduction, overloadable operators, unary operator overloading, operator keyword, binary operator overloading, concatenation of strings

Inheritance

Defining a subclass, forms of inheritance, inheritance and member accessibility, constructor and destructor in derived class, overloaded member functions, Multilevel inheritance, Multiple Inheritance, Hybrid Inheritance, Hierarchical inheritance

File I/O and Streams

Stream Classes, Character Stream, Byte Stream, Using Stream I/O, Serialization

Text Books

1. BjarneStroustrup, –C++programming language ,Third edition
2. E. Balaguruswamy, Object-Oriented Programming with C++, Tata McGraw Hill, 6th Edition 2013

Recommended Reading

3. E. Balaguruswamy, Programming with ANSI C, Tata McGraw Hill, 7th Edition2017.
4. Herbert Schildt , C++ Complete Reference, Tata McGraw Hill, 4th Edition2003.
5. RobertLafore,‖Object-OrientedProgrammingInC++‖,SamsPublishing,FourthEdition
6. T. Rajkumar, K.R Venugopal, T Ravikumar Mastering, C++ Tata McGraw Hill 1st Edition 2012.
7. Walter Savitch, –Problem solvingwithC++|,Pearson/Addison-Wesley, Sixth Edition
8. JoyceFarrell, –Object-Oriented Programming Using C++|, Thomson/CourseTechnology, FourthEdition

Programme Name	***F.Y.M.C.A***	***Semester -I***
Course Code	**R4MC5012L**	
Course Title	**C/C++ Programming Lab**	

COURSE OUTCOMES

1. Apply knowledge of computer fundamental, specify design aspects to reduce complexity of a real world problem.
2. Design and evaluate the a real world problem using basic constructs of C/C++ language to reduce the number of lines of code and complexity of code..
3. Recognize the need and ability to comprehend the dynamic memory allocation, inheritance in C/C++.
4. Develop an application using OOP concepts using modern computing tools.

TITLE OF EXPERIMENT

Note: under each of the topics 6 or more programming question with variation can be solved.

1. Write a C program to illustrate the use of operators and loops
2. Write a C program to illustrate the use of function covering all topics under function.
3. Write a C Program to illustrate arrays (one/two) dimensional, passing of arrays to the function.
4. Write a C Program to illustrate use of structures, nested structure, enumerated types, Storageclass
5. Write a C Program to handle strings, passing of strings to the function, and string manipulation.
6. Write a program to illustrate use of pointers, manipulation of pointers.
7. Write a program to illustrate use of classes and objects with use of Constructors and Destructors
8. Write a C++ Program to illustrate the use of operator overloading
9. Write a C++ Program to overload as binary operator, friend and member function
10. Write C++ Programs and incorporating various forms of Inheritance

Programme Name	***F.Y. M.C.A.***	***SEMESTER I***
Course Code	**R4MC5015T**	
Course Title	**Data Communication Network**	

PREREQUISITES

1. Design and analysis algorithms
2. High-level programming languages of C, C++ or Java
3. Fundamental concepts in operating system

COURSE OUTCOMES

1. Explain basic networking concepts and basic communication model.
2. Describe network architectures and components required for data communication.
3. Analyze the function and design strategy of physical, data link, network layer and Transport Layer.
4. Design, calculate, and apply subnet masks to fulfill networking requirements and building the skills of routing mechanisms.
5. Describe various application protocol standards developed for internet giving justification for each.

COURSE CONTENTS

Overview of Data Communication and Networking

What Is the Internet?, What Is a Protocol?, Overview of data (analog & digital), signal (analog & digital), transmission (analog & digital), Direction of data flow (simplex, half duplex, full duplex); physical structure (type of connection, topology), categories of network (LAN, MAN, WAN); The Network Edge, The Network Core, Circuit Switching, Packet Switching, and Message Switching, Access Networks, and Physical Media, Delay and Loss in Packet-Switched Networks, Protocol Layers and Their Service Models.
Internet Backbones, NAPs, and ISPs

Application Layer:

Principles of Application Layer Protocols, The World Wide Web: HTTP, File Transfer: FTP, Electronic Mail in the Internet, DNS--The Internet's Directory Service, Socket Programming with TCP, Socket Programming with UDP,

Transport layer:

Transport-Layer Services and Principles, Multiplexing and Demultiplexing Applications,

Connectionless Transport: UDP, Principles of Reliable Data Transfer, TCP features, byte-stream, connection-oriented, TCP header format, 4-tuple, incarnations of a TCP connection, 3-way handshake, initial sequence numbers, TCP state diagram, TCP sliding window, , RTT estimation, Jacobson/Karel's algorithm for RTT estimation, slow-start, congestion avoidance, self-clocking/ackclocking, TCP Tahoe, Fast retransmit, fast recovery, Resource allocation models, evaluation metrics, queuing models, quality of service Congestion control algorithm: Leaky bucket algorithm, Token bucket algorithm, choke packets; Quality of service: techniques to improve QoS.

Network Layer and Routing:

Introduction and Network Service Models, Routing Principles, Hierarchical Routing, Internet Protocol, Routing in the Internet, What's inside a Router?, IPv6, Multicast Routing,

Link Layer and Physical Layer:

The Data Link Layer: Introduction, Services, Error Detection and Correction Techniques, Multiple Access Protocols and LANs, LAN Addresses and ARP, Ethernet, Hubs, Bridges, and Switches, IEEE 802.11 LANs, PPP: The Point-to-Point Protocol, Asynchronous Transfer Mode (ATM), X.25 and Frame Relay, Point to point protocol, LCP, NCP, FDDI, token bus, token ring; Reservation, polling, concentration; Multiple access protocols: Pure ALOHA, Slotted ALOHA, CSMA, CSMA/CD, FDMA, TDMA, CDMA; Traditional Ethernet, fast Ethernet. Spatial Sorting, Level of detail, collision detection, standard objects, and physics, TDM, FDM, WDM; Circuit switching: time division & space division switch, TDM bus; Telephone network.

Text Book

1 James F. Kurose and Keith W. Ross, “Computer Networking: A Top-Down Approach” Addison Wesley, Sixth edition
2 S. Tanenbaum, Computer Network, Pearson Education, PHI, Fifth Edition

Recommended Reading

3 William Stallings,” Data and Computer Communications”, Pearson Education, Tenth Edition
4 B. A. Forouzan: Data Communications and Networking, Fifth Edition, Tata McGraw Hill Education Private Ltd
5 Alberto Leon-Garcia &Indra Widjaja, Communication Networks Fundamental Concepts and Key Architectures, TMH, 2001.

Programme Name	*F.Y. M.C.A.*	*SEMESTER-I*
Course Code	**R4MC5015L**	
Course Title	**Data Communication Network Lab**	

PREREQUISITES

Knowledge of any programming language is required for this course

COURSE OUTCOMES

1. Explore networking commands.
2. Analyze headers of various networking layers.
3. Implement the various encoding schemes, error detecting codes, various congestion control protocols and socket programming.
4. Implement client server model.
5. Design a network architecture using appropriate tools that solves real world designing problem.

TITLE OF EXPERIMENT

1. Familiarization of simple networking commands like ping, etc.
2. Socket programming – UDP
3. Using TCP/IP sockets, write a client – server program to make the client send the filename and to make the server send back the contents of the requested file if present.
4. Implement the above program using as message queues or FIFOs as IPC channels.
5. Write a program for congestion control using leaky bucket algorithm.
6. Encoding scheme (NRZ, NRZ-I, Manchester, Differential Manchester)
7. Write a program for sliding window algorithm.
8. Write a program for error detecting code using CRC (16-bits).
9. Design network architecture using wired or wireless media.

Programme Name	*F.Y.M.C.A*	*Semester I*
Course Code	**R4MC5013S**	
Course Title	**System Modelling**	

PREREQUISITES

Knowledge of C++ or any programming language is required for this course. Excellent communication skills will be an added advantage.

COURSE OUTCOMES

1. Identify and transcribe the system requirements of problems to solve simple societal problems.
2. Analyze the various modules of the system using appropriate techniques and tools.
3. Perform data modeling using various methods and available open source tools.
4. Design reliable software useful for community.
5. Understand the procedures to conduct the various meetings (in groups or one on one) and review to know the current status of the system.

COURSE CONTENTS

Introduction to Structured Analysis and Design

Concept of System Analysis and design. Information systems, classification, Requirement at different levels of management. Transaction processing system, Management Information System, Decision Support System. Determining the scope and structure of a system. Role of a System Analyst

Systems Development strategies

SDLC, Structured Analysis development method, System development methods and their Advantages and disadvantages

Understanding System Requirements:

Fact finding techniques: Interview, Questionnaire, Observation, record review. Tools for documentation: Decision tree, Decision tables, Structured English

Structured System Analysis

Components of structured analysis: Data Flow analysis, Data Dictionary

Systems Design

Specifying application requirements, Design of output, Different formats of output, Input design, Input validation, Design of files and databases

Software Development and Quality assurance

Designing reliable systems. Program structure charts, Design of software, software design and documentation tools.

Introduction to Object Oriented Modelling

An overview, Object basics, Object state and properties, Behavior, Methods, Messages, Information hiding, Class hierarchy, Relationships, Associations, Aggregations, Identity, Static and Dynamic binding, Persistence, Metaclasses, Object oriented system development life cycle.

Methodology and UML

Introduction, Survey, Rumbugh, Booch, Jacobson methods, Patterns, CRC(class responsibility and collaboration), Frameworks, Unified approach, Unified modeling language, Static and Dynamic models, UML diagrams, Class diagram, Use case diagrams, Dynamic modeling, Model organization, Extensibility.

Object Oriented Analysis

Identifying Use case, Business object analysis, Use case driven object oriented analysis, Use case model, Documentation, Classification, identifying class, Flexibility guidelines for class diagram: Cohesion, Coupling, Identifying object relationships, attributes, methods, Super-sub class, A part of relationships Identifying attributes and methods, Object responsibility.

Object Oriented Design

Design process and benchmarking, Axioms, Corollaries, Designing classes, Class visibility, Refining attributes, Methods and protocols, Object storage and object interoperability Databases, Object relational systems, Designing interface objects, Macro and Micro level processes, The purpose of a view layer interface-OOUI, MVC Architectural Pattern and Design, Designing the system. Framework and Design Patterns

Text Books

1. James A. Senn Analysis and design of Information Systems Tata McGraw Hill, 2008.
2. Ali Bahrami, Object Oriented System Development, McGraw Hill International Edition, 1999.

Recommended Reading

1. V. Rajaraman Analysis and design of Information Systems, 3rd Edition2011.
2. Richter, Charles. Designing flexible object-oriented systems with UML. New Riders Publishing, 1999.
3. Rumbaugh, James, Ivar Jacobson, and Grady Booch. Unified Modeling Language Reference Manual, The. Pearson Higher Education, 2004.

Programme Name	***F.Y.M.C.A***	***Semester I***
Course Code	**R4MC5013P**	
Course Title	**System Modelling Lab**	

PREREQUISITES

Knowledge of C is required for this course

COURSE OUTCOMES

1. Evaluate different techniques for study of existing software
2. Analyze existing system and represent using various structure modeling techniques.
3. Design the database, input forms and output forms.
4. Design the efficient input structure, output structure and storage classes for the software project
5. Create a data model.

TITLE OF EXPERIMENT

1. Description of the proposed system.
2. Fact Gathering and Feasibility study.
3. Create decision trees, Data Flow diagrams, Structure charts.
4. Create ER Diagrams.
5. Input Output Module and Table Design
6. Development of test cases and testing of software.
7. Create Training and Implementation schedule.
8. To perform the user's view analysis: Use case diagram, Class diagrams.

Programme Name	*F.Y.M.C.A*	*Semester -I*
Course Code	**R4MC5014S**	
Course Title	**Computer Organization & Architecture**	

PREREQUISITES

Knowledge of handling system and binary number system.

COURSE OUTCOMES

1. Design of Combinational / sequential Logic circuits for solving day to day problems in real world ethically.
2. Explain the role of different components of computer.
3. Write modular assembly language programs that execute given specifications, terminate normally.
4. Describe and Analyze various data transfer techniques in digital computer.
5. Compare and assess issues related to memory and control functions to ensure fault handling and reliability.
6. Understand the concepts of Micro-programmed and Hard Wired approaches, multicore processors, instruction level parallelism and other advanced concepts.

COURSE CONTENTS

Digital Principles, Digital Logic

Definitions of Digital Signals, Numeric data representations and number bases, Fixed and Floating point systems, signed and 2‘s complement representation, BCD codes, Gray Code, Excess-3 code, Error detecting and correcting codes, Conversion between codes, The Basic Gates: NOT, OR, AND, Universal Logic Gates: NOR, NAND, Positive and Negative Logic.

Combinational Logic Circuits

Basic theorems of Boolean Algebra, standard and canonical forms, Truth Table to Karnaugh Map, Karnaugh Simplifications, Don‘t-care conditions, Sum-of-Products Method, Product-of-sums Method, Product-of-sums Simplifications, Multiplexer, DeMultiplexers, Implementations of basic arithmetic and logic circuits such as Adders, Subtractors, Binary Parallel adders, Decimal Adders, Magnitude comparators, Decoders, Programmable LogicArrays.

Sequential Logic Circuits

Introduction Flip flops, Representation of FLIP-FLOPs, Triggering of flips-flops, Analysis of Sequential Circuits, Types of Registers, Universal Shift Register, Applications of Shift Registers, Asynchronous Counters, Synchronous Counters, Counter Design as a Synthesis problem, Changing the Counter Modulus.

Basic Structure of Computers

Basic organization of von Neumann machine, Instruction execution in von Neumann machines, Evolution of Computer Types, Functional and structural organization of computers, Functional Units, Basic Operational Concepts, Bus Structures, Performance – Processor Clock, Multiprocessing and Multicomputers

Machine Instructions and Programs

Memory Location and Addresses, Memory Operations, Instruction sets, Instruction formats and Instruction Sequencing, Addressing modes, Introduction to Assembly language (8085), Basic Input and Output Operations, subroutines in Assembly language

Input / Output Organization

Accessing I/O Devices (I/O fundamentals: handshaking, buffering, programmed I/O), Interrupts, interrupt driven I/O, Exceptions, Direct Memory Access, Buses (protocols, arbitration), Interface Circuits, Standard I/O Interfaces – PCI Bus, SCSI Bus, USB.

Memory Systems

Storage systems and their technology, Memory Hierarchy: importance of temporal and spatial locality, Semiconductor RAM Memories, Read Only Memories, Speed, Size, and Cost, Latency, cycle time, bandwidth and interleaving, Cache Memories – Mapping Functions, Replacement Algorithms, Performance Considerations, Virtual Memories (page table, TLB), Fault handling and reliability, Secondary Storage.

Basic Processing Unit

Some Fundamental Concepts, Execution of a Complete Instruction, Multiple-Bus Organization, Hardwired Control, Microprogrammed Control Pipelining: Basic Concepts, Data Hazards, Instruction Hazards, Influence on Instruction Sets, Data Path and Control Considerations, Super Scalar Operation, UltraSPARC 2 Example, Performance Consideration.

Large Computer Systems:

Forms of Parallel Processing, Array Processors, The Structure of General-Purpose Multiprocessors, Interconnection Networks.

Text Books

1. Stallings, William. Computer organization and architecture: designing for performance. Pearson Education India, Xth edition.
2. Jain R. P, Modern Digital Electronics, TMH, 4th Edition

Recommended Reading

1. Stallings, William. Computer organization and architecture: designing for performance. Pearson Education India,2000.
2. M. Morris, R. Mano Logic Fundamentals and Computer Design,Prentice Hall of India, 4th Edition, 2007.
3. David A. Patterson and John L. Hannessy Computer Organization and Design: The Hardware and Software Interface, Morgan Kaufmann, Elsevier, 4th Edition,2012.
4. John Hayes, Computer Architecture and Organization, Tata McGraw Hill, 5th Edition, 1996.
5. Pal Chaudhauri, Computer organization and Design Prentice Hall of India, 2nd Edition, 2000.
6. V. Rajaraman and T. Radhakrishnan Introduction to Digital Computer design Prentice Hall of India, 5th Edition,2008.

Programme Name	***First Year MCA***	***Semester I***
Course Code	**R4MC5018L**	
Course Title	**Business English**	

PREREQUISITE

COURSE OUTCOMES

2. Acquire basic proficiency in English grammar and vocabulary.
3. Develop enhanced writing skills required for technical communication.
4. 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

Summarisation, email, business letters

Speaking skills

Overcoming fear, organising 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

Programme Name	*F.Y.M.C.A*	*Semester -I*
Course Code	**R4MC5016L**	
Course Title	**Web Technology Lab**	

PREREQUISITES

Knowledge of an internet and web pages

COURSE OUTCOMES

1. Apply knowledge of computer fundamental, specific to Design static webpages.
2. Design Dynamic web pages using various component to meet the specified needs with appropriate tools.
3. Demonstrate knowledge and understanding of XHTML and XML to design a well formatted webpage.
4. Apply appropriate technique, resources and modern computing tools to design a well formatted web page.

TITLE OF EXPERIMENTs

1. Design an experiment using HTML to demonstrate the usage of
    a) Relative URLs
    b) Lists
    c) Tables
    d) Frames
    e) Images
    f) Links
    g) Fonts and colors
    h) Forms
2. Design an experiment using JavaScript to demonstrate the usage of
    a) Various branching structures
    b) Various looping structures and dialog boxes
    c) Using of cookies, function and events
    d) Exploit the various object like date, string, array
    e) To validate a form
    f) Animate an image
3. Design an experiment using Cascading style sheets to demonstrate the usage of
    a) Simple stylesheet
    b) Colors, background, Fonts
    c) Images, links, Borders, list
    d) Classes, dimensions

4. Design an experiment using XML to demonstrate the usage of

a) Simple tags
b) Create elements and its attributes
c) Xml schemas
d) Adding of DOM object
e) XML databases

5. Design an experiment using XHTML to demonstrate the usage of
   a) Simple tags
   b) Various document types
   c) Attributes, events and validations

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.	R4MC5021S	Probability and Statistics	3	1	--	4.0	20	20	60	3
2.	R4MC5022T	Core and advanced Java Programming	3	-	-	3.0	20	20	60	3
	R4MC5022L	Core and advanced Java Programming Lab	-	-	3	1.5	100% CIE			
3.	R4MC5023S	Accounting Banking and Finance	3	-	-	3.0	20	20	60	3
4.	R4MC5024T	Data and file Structures	3	-	-	3.0	20	20	60	3
	R4MC5024L	Data and File Structures Lab	-	-	2	1.0	100 % CIE			-
5.	R4MC5025T	Operating Systems	3	-	-	3.0	20	20	60	3
	R4MC5025L	Operating Systems Lab	-	-	2	1.0	100% CIE			
6.	R4MC6015S	Professional and Technical Communication	1	-	2	2.0	20	20	60	3
7.	R4MC5028P	Unix Programming Lab	-	-	2	1.0	100% CIE			-
8.	R4MC5027L	Mini Project 1	-	-	3	1.5	100 % CIE			-
		Total	16	1	14	24				

Abbreviations: **L**: Lecture, **T**: Tutorial, **P**: Practical, **TA**: Teacher Assessment / Term work Assessment, **MST**: In Semester Test, **ESE**: End Semester Written Examination, **CIE**: Continuous In-semester Evaluation

Programme Name	***F.Y.M.C.A.***	***SEMESTER II***
Course Code	**R4MC5021S**	
Course Title	**Probability and Statistics**	

PREREQUISITES

Knowledge of Mathematics and probability.

COURSE OUTCOMES

1. Apply probability concepts to solve real world problems
2. Translate variety of real-world problems into distribution models
3. Apply the concept of a sampling distribution and assess the consistency, efficiency and unbiasedness of estimators
4. Apply the concepts of statistical inference and time series analysis in practical data analysis
5. Apply the reliability theory knowledge to manage network load or software program codes.

COURSE CONTENTS

Probability

Sample spaces, events as subsets, probability axioms, finite sample spaces, binomial coefficients and counting techniques applied to probability problems, conditional probability, independent events, and Baye's formula.

Random variables and their distributions

Random variables (Discrete and continuous), probability functions, density and distribution functions

Distributions: Binomial, Poisson, geometric, hyper geometric, uniform, exponential, negative exponential, normal, mean, variance, skewness, kurtosis, Chebychev inequality, independent random variables and their distributions.

Statistical Inference

Estimation and sampling, point and interval estimates, hypothesis testing, power of a test. Regression, correlation, non parametric tests.

Statistical methods useful in practice

Tests of hypotheses, tests of measures, normality, Tests of significance: Contingency tables, Chi square test, analysis of variance.

Reliability of a system

System consisting of components having varied reliability based on that reliability of the whole system in parallel and series placement of its components.

Time series analysis

Trends and seasonal variations, exponential smoothing, forecasting techniques with focus on predictions and error analysis of results, Jenkin‘s prediction model.

Text Books

1. Gupta, S. C., and Dr VK Kapoor. Fundamentals of mathematical statistics: A modern approach. Sultan Chand, 2000.

Recommended Reading

1. Trivedi, Kishore S. Probability & Statistics with Reliability, Queuing and Computer Science Applications. PHI Learning Pvt. Limited,2011.
2. S.D. Sharma, R. K. Malhotra, Operations Research ,Anmol Publisher,1997.
3. HamdyTaha, Operations Research, 8th edition Pearson Education,2008.

Programme Name	*F.Y.M.C.A.*	*SEMESTER-II*
Course Code	**R4MC5022T**	
Course Title	**Java Programming**	

PREREQUISITES

Knowledge of object oriented programming concepts.

COURSE OUTCOMES

1. Understand the fundamental concepts of Java such as operators, classes, objects, inheritance, packages, Enumeration and various keywords..
2. Apply Object Oriented concepts to classes and objects.
3. Solve complex computational problems using strings, arrays, files and databases
4. Apply the basic principles of creating Java applications with graphical user interface (GUI), networking and Database.
5. Analyze a real world problem and break into logical modules and solve using Java programming.

COURSE CONTENTS

Introduction to Java

Java Features: Simple, Compiled and Interpreted, Platform independent and portable, Object oriented

Introduction to Object Oriented Programming

Object and Classes, Features of Object Oriented Programming, Data abstraction and encapsulation, Inheritance, Polymorphism, Dynamic Binding, Resilience to change, Reusability of Code, Modularity of Code

Data Types in Java

Data Types, Constant, Symbolic Constant, Scope of variable, Type casting, standard default values, Java Literals.

Operator and Expression

Arithmetic Operators, Bit wise Operator, Relational Operators, Boolean Logical Operators, Assignment Operator Increment and Decrement Operator, Conditional Operator, Special Operator, Operator Precedence.

Decision making and Branching Statement

Java‘s Selection Statement, if statement, if else statement, The else if ladder, The switch statement, The? : Operator, While statement, do while statement, for statement, Nested Loops, Jumps in Loops, Labeled Loops, exception handling

Array

Arrays, One Dimensional array, creating an array, Two Dimensional array, passing arrays as parameters, Recursion.

Classes, Object and Methods

Class Fundamentals, Declaring and Creating object, Accessing class, members and methods, Constructor, Methods Overloading, Static Member

Inheritance

Defining a subclass Constructor, Multilevel inheritance, Hybrid Inheritance, Hierarchical inheritance, Overriding Methods, Final variable and Methods, Final Classes, Abstract method and Classes

Packages and Interfaces

Interface: Defining interfaces, Extending interfaces, Implementing interfaces, Accessing Interface variable, Applying Interface. **Packages:** Define a Package, Using system Package, Naming Convention, Creating Package, Accessing a package, Import a package, adding a class to a package.

Visibility Control

Public access, friend access, protected access, Private access, Private Protected access

Strings, Vectors and Wrapper Class

Strings, Vectors, Wrapper Classes, garbage collector

Exception handling

Types, use of Exception handling, Exception handling constructs, try-catch-finally, throw statements, user defined exceptions

Multithreaded Programming

The Java Thread Model, The Main Thread, Creating Thread, Extending a thread class, Stopping and Blocking a thread, Life cycle of thread, Using thread method, Thread exceptions, Thread priority, Synchronization, Implementing a Runnable‘_ Interface

Graphics Programming and Internet Programming (Applets)

Applet Class, Applet Architecture, Local and remote applets, How applet differ from application, Preparing to write applets, Building applet code, Applet life cycle, Creating an Executable Applet, Designing a Web page, Applet tag, Adding Applet to HTML file, Running the Applet, Passing parameter to applet. Graphics Programming: The Graphics Class, Lines and rectangle, Circle and Ellipse, Drawing Arcs, Drawing Polygons, Line Graphs, Using control loops in Applets, Drawing Bar charts, Panel, Frame, Canvas

File I/O and Streams

Stream Classes, Character Stream, Byte Stream, Using Stream I/O, Serialization

Networking and Sockets

Networking Basics, Connection oriented communication, Connectionless communication, Communication between client and socket and server socket, Server Socket class, Socket class, Proxy Server, UDP Sockets

Java Database Connectivity

JDBC Structure, JDBC, ODBC Bridge, SQL Package Open database, ODBC, Data Source name, ODBC Structure, Database Drivers

Text Books:

1. Herbert Schildt, Java Complete Reference Tata McGraw Hill, 10th edition.

Recommended Readings

1. Rajkumar Buyya, Thamarai Selvi, Xingchen Chu, Object-Oriented Programming with Java 1st Edition Tata McGraw Hill,2004.
2. E.Balaguruswamy, Programming with Java A Primer, 5th Edition, Tata McGraw Hill 2014.
3. Jana, Debasish. Java and object-oriented programming paradigm. PHI Learning Pvt. Ltd., 2005.
4. Bruce Eckel, Thinking in Java, Fourth Edition, Pearson,2006.
5. Cay Horstmann, Big Java ,Wiley India,2008.

Programme Name	***F.Y. M.C.A.***	***SEMESTER-II***
Course Code	**R4MC5022L**	
Course Title	**Java Programming Lab**	

PREREQUISITES

Knowledge of C/C++ and object oriented programming concepts.

COURSE OUTCOMES

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

TITLE OF EXPERIMENTS:

Note: under each of the topics atleast 2-3 programming question with variation can be solved.

1. Write some Simple java Programs using various decision making and looping structures.
2. Write a java Program for creation of classes and use of different types of functions like constructor, Destructor, function overloading etc.
3. Write a java Program to count the number of objects created for a class using static member function
4. Write a java Program using interfaces and package
5. Write a java Program using inheritance
6. Write a java Program using IO streams and exception handling mechanism
7. Write a program using Files
8. Write a Programs using AWT
9. Write a Programs to implement various built in classes like String, Linked list, Arrays and Vectors
10. Write a Programs on swing.
11. Write a Programs using JDBC.

Programme Name	*F.Y. M.C.A.*	*SEMESTER-II*
Course Code	**R4MC5024T**	
Course Title	**Data and file Structures**	

PREREQUISITES

Knowledge of C and an Object Oriented language such as Java or C++

COURSE OUTCOMES

1. Use basic data structures, stack, queue, array, tree, graph as tool to solve real world problems.
2. Compare various searching and sorting algorithms.
3. Apply tree and graphs to find shortest path, minimum spanning tree, compress code, expression tree and traversal sets.
4. Compare and contrast various file organization for a given application.
5. To compare and choose right (appropriate) index structure.

COURSECONTENTS

Linear Data Structures

Stacks, Queues and its applications.

Trees

Introduction, Binary Trees, Binary Tree Traversals, Threaded Binary Trees, Heaps, Binary Search Trees

Graphs

Directed, Undirected Graphs, Matrix and Pointer representation of Graphs, Graph traversal, All paths, shortest path algorithm, Spanning Trees, Minimum spanning Trees

Searching and Sorting

Array Sorting, File Sorting, Selection and Bubble Sort, Sequential search, Search Trees, Depth First & Breadth First Search, Divide and Conquer: General Method, Binary Search, Merge Sort, Quick Sort and its performance, O _, _o‘, ‟Ω‘ notations , its relation with complexity of Algorithms, Determine the space and time complexity of various algorithms.

Hashing

Hash Function, address calculation techniques, collisions, collision handling, collision resolution.

File Structures

Definitions, concept of record, file operations: Storing, creating, retrieving, updating Sequential, relative, indexed and random access mode, Files with binary mode(Low level), performance of Sequential Files, Direct mapping techniques: Absolute, relative and indexed sequential files (ISAM) concept of index, levels of index, overflow of handling.

File Handling

File operation: creation, copy, delete, update, compare, text file, binary file.

Index Structures

Searching, Insertion, Deletion, Performance, Binary Search Tree as Index, M-way Search Tree, B Tree, B+ Tree, B* Tree

Text Books:

1. Horowitz ,Sahani , Data Structures Using C++,TMG,1999.
2. Langsam, Augestein, Tanenbaum, Data Structures using C and C++ , PHI, Second Edition, 2007.

Recommended Reading

1. Cormen, Leiserson, Rivest, Stein, Introduction to Algorithms,Third Edition, PHI,2009.
2. Mary Loomis, Data Management & File Structures, Second Edition, PHI,2009.

Programme Name	*F.Y. M.C.A.*	*SEMESTER-II*
Course Code	**R4MC5024L**	
Course Title	**Data and File Structures Lab**	

PREREQUISITES

Knowledge of an advanced C/C++

COURSE OUTCOMES

3. Implement and compare sorting algorithms
4. Implement and compare searching Algorithms
5. Implement hashing algorithms
6. Design the basic tree structure.

TITLE OF EXPERIMENTS

1. Write a program which accepts different data sets from input file & sorts those data sets using Six array sorting methods
    a) St Insertion
    b) St selection
    c) St Exchange
    d) Heapsort
    e) Quick sort
    f) Shell sort
2. The output will be written on a file in tabular format where individual entry in tale will be no. of moves & no. of comparisons made for that data set & using that sorting method.
3. Write a program for Breadth First Search
4. Write a program for Depth First Search
5. Implement Hashing techniques
6. Write a program to sort a file using various file sorting techniques
7. Write a program to build. insert on B-Tree
8. Write a program to delete from B-Tree

Programme Name	*F.Y.M.C.A*	*Semester II*
Course Code	**R4MC5025T**	
Course Title	**Operating System**	

PREREQUISITES

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

COURSE OUTCOMES

1 Design and evaluate complexity of various process management techniques.
2 Apply various disk scheduling algorithms to evaluate efficacy of the same.
3 Demonstrate knowledge and understanding of Files.
4 Use research based knowledge identify system Parameters used to evaluate system performance.
5 Recognize the need to improve the relation between tuning and Performance parameters.

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, windowing systems.

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, nonsequential), 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

Text Books:

1. William Stallings, Operating System, Internals and Design Principles, Prentice Hall, 7th edition, 2011.

Recommended Reading

1. Silberschatz,Galvin,WileyIndia,OperatingSystemConcepts,JohnWiley&Sons,7th edition, 2006.
2. 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	**R4MC5025L**	
Course Title	**Operating System Lab**	

PREREQUISITES

Good knowledge of C

COURSE OUTCOMES

1. Compare and contrast various CPU scheduling algorithms.
2. Select adapt and apply various Memory management algorithms.
3. Evaluate various Disk-scheduling algorithms.
4. Simulate resource management in OS.

TITLE OF EXPERIMENT

1. Implement CPU scheduling Algorithms
2. Implement Memory Management Algorithms
3. Implement Disk-scheduling Algorithms
4. Simulate a basic OS

Programme Name	*F.Y.M.C.A*	*Semester -II*
Course Code	**R4MC5023S**	
Course Title	**Accounting Banking and Finance**	

PREREQUISITES

Knowledge about basic mathematics.

COURSE OUTCOMES

1. Understands basic working of banks and financial markets.
2. Analyze the financial statements.
3. Extend the accounting principles to implement using softwares.
4. Describe the double entry book keeping system.
5. Prepare financial statements like balance sheet; and profit and loss account.

COURSE CONTENTS

Financial System:

Financial Institutions and Financial Markets

Banking and financial institutions in India: Commercial Banks, Cooperative Banks, Regional Rural Banks, Agriculture and Rural Development Banks (SLDBs), Development Banks and NBFCs; their constitution, Functions, Working and Evolution. Reserve Bank of India: Traditional and Developmental Functions and Working. Objectives, Instruments of Monetary Policy

Financial markets: Structure, institution and operating mechanism and its role in Economic Development. Developed and Underdeveloped Markets, Money Market in India: Importance, features instruments. Measures to strengthen money market in India. Recommendations of the working group on money market, Report of the task force on money market and mutual funds.

Capital Markets in India: New issue market and stock exchange, importance of stock exchanges: National stock exchange, methods of floating new issues, types of shares and debentures, The Securities and Exchange Board of India (SEBI), need for establishment of SEBI. Objectives and role of SEBI, Capital Market

BANK MANAGEMENT

Nature of Banking Business: Socio, Economic and legal environment of banking business in India – official regulation and control over banks in India: Banking Regulation Act, 1949, Reserve Bank of India Act, 1934, Banking companies Act, 1970 etc – Diversification of Banking in India.

Forms of Banking: Branch Banking, Unit Banking, Group Banking Chain Banking, Banking, correspondent Banking. Process of Bank Management; Branch location policies and decisions, organizational Structure of Commercial Banks in India Department set up of head office, Zonal Office, Regional office and Branchy offices. Delegation of authority in banks.
Bank Balance Sheet: Management of assets and liabilities in banks – Profit Profitability and Productivity in banks – Management of large sized branches and rural branches. Internal control and Performance budgeting system – Management Information system Income Recognition and asset classification norms Human Resource Development in Banks: Manpower Planning, Recruitment, Training, Promotion, Motivation, Bank Marketing: Product Planning and Development. Computerization of Banks: Need, application, progress, problems. Current Issues and Problems of Management of Banks in India.

Bank Credit (Banking part -2)

Establishing banks lending policies under changing socio-economic and legal environment –types and Forms of bank advances – Principles of credit management – types of documents used for lending – Modes of creating charge – Evaluation of different types of securities.
I Analysis of financial statements with the help of accounting ratios fund flow and cash flow statements and other emerging techniques Credit Appraisal Techniques: Financial analysis techniques for bank lending – Assessment of credit needs for fixed assets and working capital – Monitoring of advances – renewal and recovery of advances – Nursing of sickunits.
I Lending Schemes: Socio – Economic and poverty alleviation programmes/ Self employment schemes: DRI, IRDP, SEEUY, SEPUP, Micro Financing: Financing of self help groups by banks and NGOs – Women Entrepreneurs, Small borrowers, SSIs, Personal and Consumer loans, Housing Finance, Agriculture finance, Loan syndication, Federal Financing.
Linking credit with Government Economic Plans: Lead Bank Scheme and Service Area Approach, role of NABARD, Recommendations of CRAFICARD [Sivaraman Committee]. ACRC [Khusro Committee], Refinance and credit guarantee facilities for banks. Current Issue and Problems.

Principles of accounting

Principles of Accounting, Overall Process of accounting, and basic concepts of Debit/Credit and the rules associated with the same. Introduction to various types of business transactions rules for posting the same in books ofaccounts.

Assets and working capital

Assets – Current assets, fixed assets, Liabilities—current liabilities, other liabilities, -- owner‘s equity; trading account; accounting records and systems; control accounts and subsidiary ledger; limitations. Fixed assets and depreciation, asset acquisition, disposal, replacement depreciation, intangible assets, inventory methods, sources of working capital, funds and cash flows.

Interpreting accounts and financial statements

Use of ratios in interpreting financial statements, limitations and other methods.

Standards for control

Variable costs/ fixed costs, cost- volume- profit analysis, break even, marginal and full costing, contribution, standard costing, analysis of variance, computer accounting and algorithms.

Budgeting and forecasting

Characteristics of budget, definition, advantages, preparation, forecasting: long term/short term

Project appraisal

Methods of capital investment decision making, discounted cash flows, internal rate of return, payback, rate of return, sensitivity analysis, cost of capital.

Text Books:

1. Chandra, Prasanna. Financial management. Tata McGraw-Hill Education, 2011.
2. Stice, Stice and Diamond, Thomson / South-Western, Financial accounting - Analysis and Reporting,2009.
3. Banks and Institutional Management by Vasant Desai

Recommended Reading

1. Ciaran walsh Key Management Ratio Tata McGraw Hill Education, 3rd Edition,2011.
2. Anthony, Robert N., David F. Hawkins, and Kenneth A. Merchant. Accounting: text and cases. McGraw-Hill,2011.
3. Brealey, Richard A., et al. Principles of corporate finance. Tata McGraw-Hill Education, 2012.
4. Narayanaswamy, R. Financial accounting: a managerial perspective. PHI Learning Pvt. Ltd.,2014.
5. Banking- Theory, Law and Practice by Gordon &Natarajan

Programme Name	***M.C.A.***	***SEMESTER III***
Course Code	**R4MC6015S**	
Course Title	**Professional and Technical 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 nonverbal 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
Role of communication in information age

Speaking

a. Intonation
b. Modulation
c. Basics of public speaking
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. Tata McGraw-Hill,2005.
4. Meenakshi Raman, Business Communication PrakashSingh,OUP
5. R. Subramanian, Professional EthicsOUP

Programme Name	*F.Y.M.C.A*	*Semester II*
Course Code	**R4MC5028P**	
Course Title	**Unix Programming Lab**	

COURSE OUTCOMES

1. Describe shell variables and environment.
2. Write problems using redirection pipes, filters and traps.
3. Define and manipulate functions in UNIX.
4. Justify interactive scripts.
5. Implement shell script using loops for simple applications.
6. Run various UNIX commands on a standard UNIX/LINUX Operating system to be able to understand and handle UNIX system calls.

TITLE OF EXPERIMENT

1. Hands on basic commands of unix: ls, who, more, cp, rm, mv, mkdir, cd, pwd, calc, banner, grep, sort, who, kill, cat, man, find, set, vi, finger, gcc,vi
2. Write a shell script program to display the process attributes.
3. Write a shell script to change the priority of processes.
4. Write a shell script to change the ownership of processes.
5. Write a program to send back a process from foreground.
6. Write a program to retrieve a process from background.
7. Write a program to create a Zombie process.
8. Write a program to create a child process and allow the parent to display "parent" and the child to display "child" on the screen.
9. Write a shell script program to check variable attributes of file and processes.
10. Write a shell script program to check and list attributes of processes.
11. Shell Script program to implement read, write, and execute permissions.
12. Shell Script program for changing process priority.
13. Write a shell script to create a file to store tables from 1 to 10
14. Program to implement exit, test and let Use corn shell array
15. Manipulate string variable
16. Use the trap statement to capture signal and handle error

Programme Name	*F.Y. M.C.A.*	*SEMESTER-II*
Course Code	**R4MC5027L**	
Course Title	**Mini Project I**	

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 given system..
2. Analyze the design the given system using Structural/ Static models.
3. Analyze the design the given system using Behavioral / Dynamic models.
4. Create Behavioral diagrams using Use Case, Structural diagrams.

TITLE OF EXPERIMENT

1. Introduction to UML:UML Overview, The Nature and purpose of Models
2. Modeling Requirements(Use Cases):Capturing a System Requirement, Use Case Relationships, Use Case Overview Diagrams
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 Swimlanes), Managing Complex Activity Diagrams
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 Sequence Fragments
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, Timing Constraints.

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-Box ComponentViews
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 Web application
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).