Department of Computer Science
Islamic University of Science & Technology

Proposed
Curriculum (Choice Based Credit System)

for

Master of Computer Applications
(MCA) Two Year Programme
2020 Onwards

CBCS Course Outline MCA

Semester-I (24 Credit Semester)						
Course Code	**Course Name**	**Paper category**	**Hours / Week**			**Credits**
			L	T	P	
18 Core Credit Units						
MCA-1T1-C	Programming Concepts in C /C++	Core	2	1	4	5
MCA-1T2-C	Technical Communication	Core	2	1	4	5
MCA-1T3-C	Discrete Mathematics	Core	3	1	0	4
MCA-1T4-C	Computer Organization and Architecture	Core	3	1	0	4
6 Elective Credit Units						
Pool A: 3 Elective Credit Units						
MCA-1E1-DCE	Programming Languages & Paradigm	DCE	2	1	0	3
MCA-1E2-DCE	Management Information System	DCE	2	1	0	3
Pool B: 3 Elective Credit Units						
MCA-1E3-DCE	Web Technologies	DCE	2	0	2	3
MCA-1E4-DCE	E-Commerce	DCE	2	0	2	3
02 Bridge Course Credit (Optional)						
MCA-1B1-BC	Computer Fundamentals	Bridge Course	2	0	0	2

Semester-II (26 Credit Semester)						
Course Code	**Course Name**	**Paper category**	**Hours / Week**			**Credits**
			L	T	P	
18 Core Credit Units						
MCA-2T1-C	Data Structures	Core	2	1	4	5
MCA-2T2-C	Database Management System	Core	2	1	4	5
MCA-2T3-C	Software Engineering	Core	3	1	0	4
MCA-2T4-C	Operating System	Core	3	1	0	4
6 Elective Credit Units						
Pool A: 3 Elective Credit Units						
MCA-2E1-DCE	Advanced Computer Organization & Architecture	DCE	2	1	0	3
MCA-2E2-DCE	Operational Research	DCE	2	1	0	3
MCA-2E3-DCE	Microprocessor & Assembly Language Programming	DCE	2	1	0	3
Pool B: 3 Elective Credit Units						
MCA-2E4-DCE	Modelling & Simulation	DCE	2	0	2	3
MCA-2E5-DCE	Computer Graphics	DCE	2	0	2	3
MCA-2E6-DCE	Numerical and Statistical Computing	DCE	2	0	2	3
MCA-2E7-DCE	Java Programming	DCE	2	0	2	3
2 credit units to be taken from outside departments						

Semester-III (26 Credit Semester)						
Course Code	**Course Name**	**Paper category**	**Hours / Week**			**Credits**
			L	T	P	
18 Core Credit Units						
MCA-3T1-C	Analysis and Design of Algorithms	Core	2	1	4	5
MCA-3T2-C	Object Oriented Analysis and Design	Core	2	1	4	5
MCA-3T3-C	Data Communication & Computer Networks	Core	3	1	0	4
MCA-3T4-C	Artificial Intelligence	Core	3	1	0	4
6 Elective Credit Units						
Pool A: 3 Elective Credit Units						
MCA-3E1-DCE	Advanced Operating Systems	DCE	2	1	0	3
MCA-3E2-DCE	Distributed Databases	DCE	2	1	0	3
MCA-3E3-DCE	Bioinformatics	DCE	2	1	0	3
MCA-3E4-DCE	Soft Computing	DCE	2	1	0	3
MCA-3E5-DCE	System Programming	DCE	2	1	0	3
MCA-3E6-DCE	Natural Language Processing	DCE	2	1	0	3
Pool B: 3 Elective Credit Units						
MCA-3E7-DCE	Digital Image Processing	DCE	2	0	2	3
MCA-3E8-DCE	Linux System Programming	DCE	2	0	2	3
MCA-3E9-DCE	Data Warehousing	DCE	2	0	2	3
MCA-3E10-DCE	Open Source Technologies	DCE	2	0	2	3
MCA-3E11-DCE	Introduction to Python	DCE	2	0	2	3
MCA-3E12-DCE	Dot Net	DCE	2	0	2	3
2 credit units to be taken from outside departments						

Semester-IV(24 Credit Semester)						
Course Code	**Course Name**	**Paper category**	**Hours / Week**			**Credits**
			L	T	P	
20 Core Credit Units						
MCA-4T1-C	Theory of Formal Languages	Core	2	2	0	4
MCA-4T2-C	Big Data Analytics	Core	2	2	0	4
MCA-4T3-C	Project work	Core	0	10	10	10
6 Elective Credit Units						
Pool A: 3 Elective Credit Units						
MCA-4E1-DCE	Wireless Communications	DCE	2	1	0	3
MCA-4E2-DCE	Cloud and Grid Computing	DCE	2	1	0	3
MCA-4E3-DCE	Information Security and Networks	DCE	2	1	0	3
MCA-4E4-DCE	Compiler Design	DCE	2	1	0	3
Pool B: 3 Elective Credit Units						
MCA-4E5-DCE	Block Chain Technologies	DCE	2	0	2	3
MCA-4E6-DCE	Machine Learning/Data Science	DCE	2	0	2	3
MCA-4E7-DCE	Quantum Computing	DCE	2	0	2	3
MCA-4E8-DCE	Advanced Java	DCE	2	0	2	3

Semester - I

Semester-I (24 Credit Semester)						
Course Code	**Course Name**	**Paper category**	**Hours / Week**			**Credits**
			L	T	P	
18 Core Credit Units						
MCA-1T1-C	Programming Concepts in C /C++	Core	2	1	4	5
MCA-1T2-C	Technical Communication	Core	2	1	4	5
MCA-1T3-C	Discrete Mathematics	Core	3	1	0	4
MCA-1T4-C	Computer Organization and Architecture	Core	3	1	0	4
6 Elective Credit Units						
Pool A: 3 Elective Credit Units						
MCA-1E1-DCE	Programming Languages & Paradigm	DCE	2	1	0	3
MCA-1E2-DCE	Management Information System	DCE	2	1	0	3
Pool B: 3 Elective Credit Units						
MCA-1E3-DCE	Web Technologies	DCE	2	0	2	3
MCA-1E4-DCE	E-Commerce	DCE	2	0	2	3
02 Bridge Course Credit (Optional)						
MCA-1B1-BC	Computer Fundamentals	Bridge Course	2	0	0	2

Course Title:	Programming Concepts in C/C++	**Course Code:**	MCA-1T1-C
Semester:	1st	**Paper Type:**	Core
Credits:	05	**Max Marks:**	125
Pre Requisite:	--------	**Co-Requisite:**	MCA-1T3-C

Marks Distribution: (Mid Term:30, End Term:50, Lab:25, Viva:10, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *The course is designed to provide complete knowledge of C language. Students will be able to develop logics which will help them to create programs, applications in C. Also by learning the basic programming constructs they can easily switch over to any other language in future.*
- *Develop an in-depth understanding of functional, logic, and object-oriented programming paradigms.*
- *To practice the fundamental programming methodologies in the C/C++ programming language via laboratory experiences.*
- *To code, document, test, and implement a well-structured, robust computer program using the C/C++ programming language.*
- *Perform object oriented programming to develop solutions to problems demonstrating usage of control structures, modularity, I/O and other standard language constructs.*
- *Demonstrate adeptness of object oriented programming in developing solutions to problems demonstrating usage of data abstraction, encapsulation, and inheritance.*

COURSE CONTENT:

UNIT I

C programming language: Evolution, Features & Importance. Basic Structure of C programs, Character Set, Identifiers, Reserved Words, Data Types, Constants, Variables, Symbolic Constants, Casting and Standard Libraries. Logical and Control Structures: Assignment, Arithmetic, Relational, Logical, Compound, Increment, Decrement, Bitwise Operators & Special Operators. IF, IF – ELSE, Nested IF – ELSE, ?: , SWITCH CASE. Looping Constructs: FOR, WHILE, DO-WHILE, EXIT, BREAK, CONTINUE

Arrays: Types of arrays, Initialization, dynamic arrays. Character Arrays & Strings. String-handling functions.

UNIT II

Functions: Concepts, Elements, Prototypes & Types. Passing Arrays to Functions. Storage classes, Recursion. Command-line arguments. Multifile programming. Preprocessing. Pointers: Concepts, Variables, swapping data, swapping address v/s data, pointers & arrays,

Pointers to pointers, pointer to strings, pointer arithmetic, additional operators, pointers to functions, void pointers.
Structures and Unions: Syntax & use, members, structures & pointers, array of structures, structures & functions, structure within structures.

UNIT III

OOPS: Evolution and need of C++, Advantages over Procedural programming. Introduction to classes and objects, Basic OOPS programming
C++ Functions: passing arguments to functions, returning values from functions, reference arguments, inline functions, default arguments, object as function argument, returning objects from functions. Constructors and Destructors, Copy Constructors

UNIT IV

Inheritance and Polymorphism: Inheritance and types, Polymorphism (static and dynamic), function overloading, function overriding, virtual functions & operator overloading.
Files: File processing in C & C++. Templates: Concepts, Function & Class templates, Standard Template library: Containers, Algorithms, Iterators and Function objects.

COURSE OUTCOMES:

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

- *Formulate simple algorithms for arithmetic and logical problems, translate the algorithms to programs (in C language),*
- *Test and execute the programs and correct syntax and logical errors,*
- *Develop efficient algorithms for solving a problem and implementation.*
- *Use the various constructs of a programming language viz. conditional, iteration and Recursion.*

Text Books:

1. Programming in ANSI C 6th Edition "E. Balaguruswamy"
2. Robert Lafore, "Object Orientation with C++ Programming", Waite Grou

References:

1. Object Oriented Programming with C++ " E. Balagurusamy"
2. Herbert Schildt, "C++ The Complete Reference", Tata McGraw Hill
3. Dennis Richie & Kernighan, "C Programming Language", Prentice Hall
4. Dietel & Dietel, "How to program", Pearson Education

Course Title:	Technical Communication	**Course Code:**	MCA-1T2-C
Semester:	1st	**Paper Type:**	Core
Credits:	05	**Max Marks:**	125
Pre Requisite:	--------	**Co-Requisite:**	----------

Marks Distribution: (Mid Term:30, End Term:50, Lab:25, Viva:10, Assignment / Presentations:10)

COURSE OBJECTIVES:

The objective of studying Technical Communication course is that it enhances the employability skills and performance at workplace. It provides discussion on modern media tools for enriching presentation skills for preparing PowerPoint slides. Thus acts as a reference for training programs offered by business houses and Industries.

COURSE CONTENT:

UNIT I

Technical Communication: Basics of Technical Communication, Barriers to communication and Technology in Communication. Communication in the Workplace: Problem Solving in Workplace Communication, Human factors in the communication failure. Guidelines for ethical communication. Active Listening: Introduction, types of listening, traits of a good listener, Active vs. Passive listening and implications of good listening.

UNIT II

Introduction to Effective Presentation strategies: defining purpose, analysing audience and locale, organizing contents, preparing outline, visual aids, understanding nuances of delivery, kinesics, proxemics, paralinguistic, chromatics. Interviews: introduction, objectives, types of interviews and job interviews, guidelines for surviving a job interview. Group Communication: Introduction, Group discussion, Organizational Group discussions and meeting conferences.

UNIT III

Paragraph Development: Central Components of a paragraph, length of a paragraph ad techniques for paragraph development. The art of condensation, steps for effective précis writing, samples and guidelines.

Reading Comprehension, purpose and reading rate, reasons for poor comprehension, improving comprehension skills, techniques for good comprehension. Memo reports: Purpose of memo reports, elements of a usable memo, interpersonal considerations in writing a memo, common types of memo report.

UNIT IV

Letters and Employment correspondence: Application letters and business correspondence, How applicants are screened for personal qualities, electronic job Hunting. Technical Proposals: Definition, purposes, Types, Characteristics, elements of structure, style and Appearance, evaluation.
Research paper: Introduction, Research paper, Dissertation, Thesis.

COURSE OUTCOMES:

At the end of the course, students will demonstrate proficiency by:

- *Understanding the characteristics of technical writing and the importance of purpose, audience, and genre for written communication in technical fields.*
- *Articulating complex engineering ideas appropriate for targeted audiences.*
- *Planning, drafting, revising, editing, and critiquing technical and professional documents through individual and collaborative writing.*
- *Writing effective technical and business documents that are grammatically and stylistically correct.*
- *Preparing and delivering professional technical presentations through applying principles of effective oral communication and slide design.*
- *Applying principles for the visual display of quantitative information.*
- *Researching, analyzing, synthesizing, and applying information to create technical reports.*
- *Recognizing ethical implications of technical communication in professional contexts.*
- *Understanding the contemporary issues in engineering from an environmental, societal, economic, and global perspective.*

Text Books:

1. William Pfeiffer, Padmaja,"Technical Communication A Practical Approach", Pearson Education.

References:

1. Meenakshi Raman & Sangeeta Sharma,"Technical Communication",Oxford University Press

Course Title:	Discrete Mathematics	**Course Code:**	MCA-1T3-C
Semester:	1st	**Paper Type:**	Core
Credits:	04	**Max Marks:**	100
Pre Requisite:	--------	**Co-Requisite:**	MCA-1T1-C

Marks Distribution: (Mid Term:30, End Term:50, Viva:10, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To familiarize students about set theory concepts, operations and uses in computer sciences.*
- *To explain concept of Discrete Structures which include Semigroups, Monoids, and Groups together with their uses in computer science and their corresponding structures in computer science*
- *To explain students concepts like Partially Ordered Sets, Lattices and Finite Boolean Algebra and use in Computer Science.*
- *Give student a knowledge about finite state Machines and possible modeling of process using finite state Machine concepts.*

COURSE CONTENT:

UNIT I

Proposition, Logic, Truth tables, Propositional Equivalence, Logical Equivalence, Predicates & Quantifiers, Sets: operations on sets, Computer representation of sets, Functions: Domain, Range, One-to-One, Onto, Inverses & Composition, Cardinality of a Set, sequences & summations, The growth of functions . Methods of Proof: Different methods of proof, Direct Proof, Indirect Proof, Mathematical Induction for proving algorithms.

UNIT II

Discrete probability, Advanced Counting Techniques: Inclusion-Exclusion, Applications of inclusion-exclusion principle, recurrence relations, solving recurrence relation. Relations: Relations & their properties, Binary Relations, Equivalence relations, Diagraphs, Matrix

representation of relations & digraphs, Computer representation of relations & digraphs, Transitive Closures, Warshall's Algorithm.

UNIT III

Partially Ordered Sets (Posets), External elements of partially ordered sets, Hasse diagram of partially ordered set, isomorphic ordered set, Lattices: Properties of Lattices, complemented Lattices. Graph theory: Introduction to graphs, Graph Terminology Weighted graphs, Representing Graphs, Connectivity of Graphs: Paths & Circuits, Eulerian & Hamiltonian Paths, Matrix representation of graphs. Graph Coloring. Cyclometric Complexity

UNIT IV

Trees: Rooted trees, Application of trees: Binary Search Trees, Decision Trees, Prefix Codes, Tree traversal, trees & sorting, spanning trees, minimal spanning trees. "Catalans Series (Number), B+ Trees, Red-Black Trees Finite Boolean algebra, Functions on Boolean algebra, Boolean functions as Boolean polynomials. Groups & applications: Subgroups, Semigroups, Monoids, Product & quotients of algebraic structures, Isomorphism, Homomorphism

COURSE OUTCOMES:

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

- *Develop mathematical and logical thinking*
- *Express real life problems it in terms of predicates, quantifiers, and logical connectives and obtained its solution*
- *Utilize the concepts of relations and functions to solve simple real life problems.*

Text Books:

1. KENNETH H. ROSEN "Discrete Mathematics & Its Applications, Tata McGraw – Hill
2. Ralph P. Grimaldi, Discrete And Combinatorial Mathematics An Applied Introduction

References:

1. LIU "Elements of Discrete Mathematics " Tata McGraw Hill
2. SCHAUMS "Discrete Mathematics" Tata McGraw Hill
3. KOLMAN/REHMAN "Discrete Mathematical Structures" Pearson Education
4. NICODEMI "Discrete Mathematics" CBS

Course Title:	Computer Organization and Architecture	**Course Code:**	MCA-1T4-C
Semester:	1st	**Paper Type:**	Core
Credits:	04	**Max Marks:**	100
Pre Requisite:	--------	**Co-Requisite:**	MCA-1T1-C

Marks Distribution: (Mid Term:30, End Term:50, Viva:10, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To know and understand the main components of a computer system and the considerations in their design.*
- *To acquire tools for comparison among alternatives.*
- *To know and understand performance measures, as well as their impact on system architecture.*
- *To understand the interplay among system components, design trade-offs, etc.*

COURSE CONTENT:

UNIT I

Fundamental concepts of computer architecture & organization, Register Organization. Interconnection Structures, Bus Interconnections, Integer/Floating Point Arithmetic & Representation, Instruction Cycle & Interrupts. Instruction Set Characteristics & Functions. Addressing Modes & Formats. Processor Organization: ALU, Design of Arithmetic Circuit, Design of Logic Circuit & Design of ALU.

UNIT II

Control Organization: Hardwired / Micro-Programmed Control, Control Memory, Address Sequencing, Design of Control Unit & Micro-Program Examples. Memory Hierarchy, Main Memory: RAM/ROM Chips. Memory Address Map, Memory Connection to CPU, Associative Memory. Hardware Organization – Match Logic, Read/Write Operation, Cache Memory,

Virtual Memory, Memory Management – Associated Hardware, I/O Organization. Peripheral Devices: I/O Interfaces, Asynchronous Data Transfer, Modes of Transfer, Direct Memory Access.

UNIT III

Microprocessor evolution & types, the 8086/8088 microprocessor family-Overview, 8086 internal architecture & software design of 8086/88, Memory address space & data organization, register circuitry, memory segmentation: generating memory address. 8086/88 minimum & maximum mode. System clock, bus cycle & unit states,

Hardwired organization of memory address space, read & write bus cycles, I/O data transfers & instruction, I/O bus cycles. Pin out diagram of 8086 microprocessor.

UNIT IV

Introduction to Parallel Processing, Basic Parallelization Techniques. Pipelining – Arithmetic & Instruction Pipelining, Vector & Array Processors. RISC, CISC

COURSE OUTCOMES:

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

- *Get familiarized with basics of computer hardware and how software interacts with computer hardware.*
- *Understand how computers represent and manipulate data, computer arithmetic and conversion between different number systems.*
- *Understand how Boolean algebra is related to designing computer logic, through simple combinational and sequential logic circuits.*
- *Understand basics of Instruction Set Architecture (ISA).*
- *Understand with a simple computer with hardware design including data format, instruction format, instruction set, addressing modes, bus structure, input/output, memory, Arithmetic/Logic unit, control unit, and data, instruction and address flow.*
- *Design combinational and sequential logic circuits, flip-flops, counters, shift registers, adders, subtractor, multiplexer, demultiplexer, Arithmetic/Logic unit.*
- *Understand concept of memory unit and input/output architecture.*

Text Books:

1. Morris Mano, "Computer System Architecture", PHI.
2. W. Stallings, "computer organization & architecture".

References:

1. V.C. Hamacher, A.G. Vranesic & S.G. Zaky, "Computer Organization", Tata McGraw Hill
2. J.P Hayes, "Computer Architecture & Organization", Tata Mcgraw Hill
3. M. J. Flynn, "Computer Architecture", Narosa
4. David A. Patterson, John L. Hannessey, " Computer Organization"
5. Govainda Rajalu, "Computer Architecture & Design" TMH

Course Title:	Programming Languages & Paradigm	**Course Code:**	MCA-1E1-DCE
Semester:	1st	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	--------	**Co-Requisite:**	MCA-1T1-C

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To introduce the major programming paradigms, and the principles and techniques involved in design and implementation of modern programming languages.*
- *To introduce notations to describe syntax and semantics of programming languages.*
- *To analyze and explain behavior of simple programs in imperative languages using concepts such as binding, scope, control structures, subprograms and parameter passing mechanisms.*
- *To introduce the concepts of ADT and object oriented programming for large scale software development.*
- *To design and extend operational and denotational definitions for basic programming language constructs.*
- *To prove properties of programs by various formal means, including structural and fix point induction.*

COURSE CONTENT:

UNIT I

The role of Programming Languages: Towards Higher Level Languages programming paradigms, Language implementation. Language Description: Syntactic Structures, Expression Notations, Abstract Syntax trees, Lexical Syntax. Data Representation: The role of types, basic types, arrays, unions & variant records, Sets, Pointers, Two String tables, types

& error checking. Procedure Activations: Introduction to Procedures, parameter passing methods, nested scope in source text, activation records, lexical scope: procedures as in C.

UNIT II

Object oriented Programming: class declarations in C++, dynamic allocation in C++, Information hiding. Functional Programming : Language of expressions , types, values & operations , approaches to expression evaluation, lexical scope, type checking, Function declaration by cases , Functions as first-class values, Implicit types, data types exception handling. Introduction to Prolog, data structures in Prolog, Programming techniques, controls in Prolog, Cuts

UNIT III

An introduction to concurrent Programming: Parallelism in hardware, Streams: implicit synchronization, concurrency as interleaving, Liveliness properties, safe accesses to shared data concurrency in ADA. Language Description: Semantic Methods, Synthesized attributes, Attribute grammars, natural semantics, Denotational Semantics.

COURSE OUTCOMES:

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

- *Develop a greater understanding of the issues involved in programming language design and implementation*
- *Develop an in-depth understanding of functional, logic, and object-oriented programming paradigms*
- *Implement several programs in languages other than the one emphasized in the core curriculum (Java/C++)*
- *Understand design/implementation issues involved with variable allocation and binding, control flow, types, subroutines, parameter passing*
- *Develop an understanding of the compilation process*

Text Books:

1. Ravi Sethi," Programming Languages ,Concepts & Constructs", Pearson Education
2. Freidman, Wand, Haynes, "Essentials of Programming Languages", PHI.

References:

1. Concepts of Programming Languages Robert .W. Sebesta 8/e, Pearson Education, 2008.
2. Programming languages – Watt, Wiley Dreamtech, 2004.
3. Programming Languages – Louden, Second Edition, Cengage,2003.
4. Programming languages – Ghezzi, 3/e, John Wiley, 1998.
5. Programming Languages Design and Implementation – Pratt and Zelkowitz, Fourth Edition
6. PHI/Pearson Education, 2001.

Course Title:	Management Information System	**Course Code:**	MCA-1E2-DCE
Semester:	1st	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	--------	**Co-Requisite:**	MCA-1T2-C

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

The objective of Course Management Information System is to bring a systematic Knowledge of Management Information technology so that it can be appreciated and understood for application in business and industry, thus more emphasis is laid to application of information to business management and thus provides communication with employees, maintain employee records and coordinate work activities.

COURSE CONTENT:

UNIT I

Overview of Management Information Systems: Introduction, Concepts & characteristics, Components of MIS, Role pf MIS, Concepts of Data, Information, Knowledge & Intelligence; Framework for Understanding MIS: Robert Anthony's hierarchy of management, MIS: Support to management, Relatedness of MIS with management activities, Attributes of Information system, Relationship between Organization & Information systems.

UNIT II

Introduction to decision making: Structured versus unstructured Decisions, Managerial Decision Making Process, Types of Decisions, Simons Model of decision making, Decision Support Systems: Overview, Concepts, Characteristics & Components, Overview of MBMS

UNIT III

Expert system: Basic Concepts, Comparison of conventional & expert systems, Structure of expert system, "Comparison of ES and DSS." Executive support system: Needs & characteristics, Role of executive support system in organization,

COURSE OUTCOMES:

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

- *Understand the scope & importance of Management Information Systems*
- *Understand the concept and applications of DBMS, Systems Engineering Analysis and Design*
- *Understand & Apply Decision Support Systems Models with Digital Firm*

Text Books:

1. Laudon, "Management Information Systems", Pearson
2. Jawadekar, "Management Infromation Systems",

References:

1. Kroenke, Management Information Systems", Mc-Graw Hill.
2. Mudrick R.G., Ross, J.E. andGleggt, J.R."Information Systems for Modern Management", PHI.
3. Jayashankar: Decision Support Systems, Mc-Graw Hill.
4. Stephen Haag & Maeve Cummings, "Information Systems Essentials", McGraw-Hill
5. Ralph Stair, "Principles of Information Systems", ISBN: 0619064897

Course Title:	Web Technologies	**Course Code:**	MCA-1E3-DCE
Semester:	1st	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	--------	**Co-Requisite:**	MCA-1T1-C

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To provide knowledge on web architecture, web services, client side and server side scripting technologies*
- *To focus on the development of web-based information systems and web services.*
- *To design interactive web pages using HTML and Style sheets.*
- *To provide skills to design interactive and dynamic web sites.*
- *To provide knowledge for implementing web applications using JavaScript.*

COURSE CONTENT:

UNIT I

Web Essentials: Clients, Servers, and Communication. The Internet-Basic Internet Protocols, Introduction, Concept of Internet- History of Internet, Protocols of Internet, World Wide Web,Web browsers, web servers, MIME, URL, HTTP , URL, Web Clients ,Web Server , Web sites, Web Applications, types of Web applications,HTTP request-response message.

UNIT II

HTML- Introduction, History of HTML, Versions-Basic XHTML Syntax and Semantics, Structure of HTML Document: HTML Tags, open and closed tags, HTML Editors, Elements, Attributes, Styles, Formatting, working with formatting, form, tables, fames, links, images, lists, multimedia: Audio, video, Graphics: Canvas and SVG, HTML APIs: Geolocation, Drag/Drop, web storage.

UNIT III

Cascading Style Sheets: Introduction, syntax, selectors, colors, backgrounds, borders, Margins, padding, outline, text, Fonts, Icons, Lists, Tables, Display, Max-width, position, CSS Responsive: Viewport, Grid-view, media Queries, Images, video, frameworks and templates. Introduction to Javascript, controls statements, Arrays and functions, pattern matching, Element Access, Event Handling

COURSE OUTCOMES:

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

- *Understand & Apply (X)HTML(5)+CSS programming*
- *Demonstrate dynamic webpage development using java script and DHTML*
- *Design a well formed / valid XML document*
- *Create a server side ASP application using database*

Text Books:

1. HTML5 Black Book by Dreamtech
2. HTML & CSS: Design and Build Web Sites by Jon Duckett,Wiley
3. Web Programming By Chris Bates , Wiley Publications

References:

1. Mastering HTML, CSS & Javascript Web Publishing by Laura Lemay, BPB Publications
2. HTML 5 Black Book, Covers CSS 3, JavaScript, XML, XHTML, AJAX, PHP and jQuery, by Dreamtech Press
3. HTML & CSS: The Complete Reference, Fifth Edition by Thomas Powell, McGraw Hill Education

Course Title:	E-Commerce	**Course Code:**	MCA-1E4-DCE
Semester:	1st	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	--------	**Co-Requisite:**	MCA-1E3-DCE, MCA-1E4-DCE

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To provide basic concepts of e-business and e-commerce, including presentation and discussion of the strategies and technologies involved.*
- *To provide theoretical and practical issues of conducting business over the internet and the Web*
- *To present methods for evaluating user needs.*
- *To provide an understanding of E-business Infrastructure, Selling and Marketing on the Web, Web Server Hardware and Software, Business-to-Business strategies, Virtual Communities, Web Portals, E-commerce Software, Payment systems, Security and User Experience.*

COURSE CONTENT:

UNIT I

E-Commerce (Introduction and Definition), Goals of E-Commerce, Technical Components Functions, Advantages and Disadvantages, Applications.

The Internet and WWW - Evolution of Internet, Domain Names and Internet Organisation (.edu, .com, mil-gov, .net ete), Internet Service provider.

UNIT II

E commerce business models: key elements of business model, Business to Consumer (B2C) model, B2B model, consumer to consumer model (C2C).Building E commerce- system development life Cycle, choosing software and hardware e commerce, site tools. Benefits of website, Registering a Domain Name, Web promotion. Internet Security, Secure Transaction, Computer Crime (types of Crimes), Threats.

UNIT III

Electronic Data Interchange, Introduction, Concepts of EDI and Limitation, Application of EDl, Disadvantages of EDI, EDI model, Electronic payment System, Introduction, Types, Strategies for developing electronic commerce web sites, Net marketplaces-characteristics of net marketplaces, types of net marketplaces, E distributors, E procurement, Exchanges. Online content providers-digital copyrights and electronic publishing

COURSE OUTCOMES:

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

- *Understand the basic business management concepts, technical concepts, legal issues, and privacy relating to E-commerce.*
- *Understand how E-commerce is affecting business enterprises, governments, consumers and people in general.*
- *Describe the infrastructure needed for E-commerce and various electronic payment systems.*

Text Books:

1. E-Commerce Concepts, Models, Strategies by G.S.V Murthy
2. E commerce: by Laudon

References:

1. E-Commerce by Kamlesh K Bajaj and Debjani Nag
2. Electronic Commerce by Gary P. Schneider

Course Title:	Computer Fundamentals	**Course Code:**	MCA-1B1-BC
Semester:	1st	**Paper Type:**	BC
Credits:	02	**Max Marks::**	50
Pre Requisite:	--------	**Co-Requisite:**	MCA-1E3-DCE, MCA-1E1-DCE

Marks Distribution: (Mid Term:15, End Term:25, Viva:05, Assignment / Presentations:05)

COURSE OBJECTIVE:

The course introduces the students to basic computer concepts. Emphasis of the course is on providing the students with an introduction to programming, programming paradigms, database, basic networking and security. The course also covers digital logic design intended to make students familiar with different types of designs as sequential logic circuits and combinational logic circuits. Widely used software's including word processing, spreadsheets and presentations are studied. Main objectives of the course are to build an appreciation for the fundamental concepts of computers and to become familiar with popular PC productivity software.

COURSE CONTENT:

UNIT I

Computers: History, Generations & Classification. Structure of a Computer System: Basic Components & Block Diagram. I/O devices & Storage Devices. H/W & S/W Concepts, Transforming data into information. Number System. Logic Gates, Boolean Algebra & K-Map. Combinational circuits, Sequential circuits, & Flip Flops. Digital Components: Integrated Circuits, Multiplexers/ Demultiplexers. Operating System: Overview, functions, features & types, Overview of Different Operating Systems. Introduction Disk Operating System (DOS) & Windows. Understanding DOS prompt, working with DOS commands, Config.sys & Autoexec.bat files.

UNIT II

Data Communications & Networking: Overview, features & types. Internet & WWW: Overview, importance, features & applications (Sharing, Browsers, E-Mails, Attachments, Search Engines, & Group Communications).
File Systems: Concepts & types. Databases: Overview, features & types. Programming languages & paradigms: Overview, features & types. Flowcharting. Control structures: conditional, looping & branching logic. Errors & their types. Introduction to Office Tools: Fundamentals of MS-Word, MS-Excel, MS-PowerPoint. PC Management: Disc Management Tools, PC tools, Norton utilities, Disk Doctor. Introduction to Computer Security: Types of infections, Viruses & Bombs, Virus Detection, Prevention & Cure Utilities.

COURSE OUTCOMES:

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

- *Understand the basic concepts of computer and its component*
- *Relate theory and practice of computer architecture*
- *Illustrate the basic data representation in the computer*
- *Comprehend the organization of the Control unit, Arithmetic and Logical unit, Memory unit and the I/O unit*

Text Books:

1. P. K. Sinha , "Computer Fundamentals, 2005", BPB, New Delhi.
2. Peter Norton, "Introduction to computers", TMH

References:

1. Taxali, "PC Software, 2005", Tata McGraw Hills, New Delhi.
2. Suresh K. Basandra,"Computers Today, 2005", Galgotia Publications.
3. Peter Norton, "Inside the PC, 2001", SAMS Tech Media.

4. Sanjay Saxena, “MS Office for Everyone, 2005”, Vikas Publications.
5. Peter Dyson, “Understanding PC Tools”, AET Publications.
6. Peter Dyson, “Understanding Norton Utilities”, AET Publications.
7. V. RajaRaman, ”Introduction to computers”, TMH

Semester - II

Semester-II (26 Credit Semester)						
Course Code	**Course Name**	**Paper category**	**Hours / Week**			**Credits**
			L	T	P	
18 Core Credit Units						
MCA-2T1-C	Data Structures	Core	2	1	4	5

MCA-2T2-C	Database Management System	Core	2	1	4	5
MCA-2T3-C	Software Engineering	Core	4	0	0	4
MCA-2T4-C	Operating System	Core	4	0	0	4
6 Elective Credit Units						
Pool A: 3 Elective Credit Units						
MCA-2E1-DCE	Advanced Computer Organization & Architecture	DCE	2	1	0	3
MCA-2E2-DCE	Operational Research	DCE	2	1	0	3
MCA-2E3-DCE	Microprocessor & Assembly Language Programming	DCE	2	1	0	3
Pool B: 3 Elective Credit Units						
MCA-2E4-DCE	Modelling & Simulation	DCE	2	0	2	3
MCA-2E5-DCE	Computer Graphics	DCE	2	0	2	3
MCA-2E6-DCE	Numerical and Statistical Computing	DCE	2	0	2	3
MCA-2E7-DCE	Java Programming	DCE	2	0	2	3
2 credit units to be taken from outside departments						

Course Title: Data Structures **Course Code:** MCA-2T1-C
Semester: 2nd **Paper Type:** Core
Credits: 05 **Max Marks::** 125
Pre Requisite: MCA-1T1-C, MCA-1T3-C **Co-Requisite:** MCA-2T2-C
Marks Distribution: (Mid Term:30, End Term:50, Lab:25, Viva:10, Assignment / Presentations:10)

COURSE OBJECTIVE:

The objective of this course is to introduce the students to the significant topics of basic as well as advanced data structures and subsequently make them understand & learn the commonly used data structures along with their applications. The focus is also to appreciate the need and working of different ways of storing data and demonstrate the advantages and disadvantages of specific data structures both linear as well as non-linear. The course emphasizes on lab work wherein the students learn not only to make use of different data structures, but also their application in different synthetic problems. The practical demonstration for autonomous realization of simple programs or program parts

is to understand the behavior of these basic data structures and to have a thorough understanding of how data structures influence the performance of algorithms.

COURSE CONTENT:

UNIT I

Data Structures: Overview & Significance. Introduction & Analysis of Algorithms. Linear & Non Linear Data Structures. Array & Strings, Application of Arrays, Sparse Matrix, Searching & Sorting: Linear & Binary Search, Bubble Sort, Selection Sort & Insertion Sort. Recursion & Applications.

UNIT II

Stack: Operations & Applications, Polish Expressions Queue: Operations & Applications, Circular Queue: Operations & Applications. Linked List: Singly Linked List, Doubly Linked List, Circular Linked List. Linked implementation of Stack & Queue, Operations & Applications of Linked Lists.

UNIT III

Trees: Basic Concept, Terminology & Representation of a Tree. Binary Tree, Binary Search Tree, Tree Traversal Techniques, BST Sort, AVL Trees, Heap Sort, Quick Sort, Merge Sort. Threaded Binary Tree, Red Black Trees, B Trees, B* Trees, Applications of Trees.

UNIT IV

Graph: Matrix & List Representation, Elementary Graph Operations (BFS & DFS), Single Source Shortest Path, Dijkstras Algorithm, Bellman-Ford Algorithm. All Pair Shortest Paths, Transitive
Closer, Floyd-Warshall Algorithm. Spanning Trees: Minimum Cost Spanning Trees, Prims & Kruskal Algorithm. Hashing: Overview, Hashing Functions, Collision Resolution Techniques.

COURSE OUTCOMES:

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

- *Develop some simple applications, like a desk calculator using stacks.*
- *Understand advanced searching methods like B-tree, B+ tree, AVL/red-black trees.*
- *Learn to develop a basic file system.*

- *Use standard libraries for data structures.*

Text Books:

1. Data Structures, Algorithms and Applications in C++, Sartaj Sahni, 2nd Ed.

References:

1. Data structures and Algorithms in C++ -- by Adam Drozdek (1994 2001).
2. Fundamentals of Data Structures in C -- by Horowitz, Sahni and Anderson-Freed (Silicon Press 2007).
3. Data Structure Using C and C++ -- by Y. Langsam, M. J. Augenstein and A. N. Tanenbaum (Pearson Education, 2nd Edition, 2015).

Course Title:	Database Management System	**Course Code:**	MCA-2T2-C
Semester:	2nd	**Paper Type:**	Core
Credits:	05	**Max Marks::**	125
Pre Requisite:	MCA-1T1-C, MCA-1T3-C	**Co-Requisite:**	MCA-2T1-C

Marks Distribution: (Mid Term:30, End Term:50, Lab:25, Viva:10, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To understand the role of a database management system in an Organization.*
- *To understand basic database concepts including the structure and Operation of the relational data model.*
- *To construct simple and moderately advanced database queries using Structured Query Language (SQL).*
- *To understand and successfully apply logical database design principles, including E-R diagrams and database normalization.*

COURSE CONTENT:

UNIT I

Basic Concepts and Conceptual Database Design: Database Users, Characteristics of the Database, Advantage of using Database Systems, Data Models, schemas and instances, Three Tier Architecture & Data Independence, Database Languages & Interfaces. Overview of Legacy Data Base Management Systems. Data Modeling Using The Entity-Relationship Model – Entities, Attributes and Relationships, Cardinality of Relationships, Strong and Weak Entity Sets, Translating your ER Model into Relational Model.

UNIT II

Relational Model, Languages & Systems: Relational Data Model, Relational Model Concepts, Relational Model Constraints, Relational Algebra, SQL – A Relational Database Language, Data Definition & Manipulation in SQL, Queries in SQL, Specifying Constraints in SQL, Practicing SQL commands using ORACLE, "PL/SQL, Stored Procedures".

UNIT III

Functional Dependencies & Normalization for Relational Databases: Functional Dependencies, Normal Forms based on primary keys, General Definitions of Second and Third Normal Forms, Boyce-Codd Normal Form, Multivalued Dependencies.
"Mini Project: Data Analysis and Data Modelling"

UNIT IV

Transaction Management: Transaction Concept and State, Desirable Properties of a Transaction, Characterizing Schedules based on Serializability and Recoverability, Concurrency Control Techniques: Lock-Based Protocols, Timestamp-based Protocols, Validation–based Protocols. Database Recovery Techniques: Recovery Concepts, Recovery based on Deferred Update and Immediate Update. Shadow Paging. Overview of Object Oriented Database Management Systems, Distributed Data Base Management Systems.

COURSE OUTCOMES:

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

- *Introduces the role of a database management system, basic database concepts, including the structure and operation of the relational data model.*
- *Familiarize themselves with the concepts of integrity constraints, relational algebra, relational domain & tuple calculus, data normalization.*
- *Construct simple and moderately advanced database queries using Structured Query Language (SQL).*
- *Have knowledge of database transaction including concurrency control, backup and recovery, and data object locking.*
- *Design and implementation of a small database project using Oracle.*

Text Books:

1. Korth, Silberschatz, “Database System Concepts”, TMH
2. Elmsari & Navathe, “Fundamentals of Database Systmes”, A. Wesley
3. Ullman J. D., “Principles of Database Systems”, Galgotia Publications

References:

1. Steve Bobrowski, “Oracle 8 Architecture”, TMH
2. Date C. J., “An Introduction to Database Systems”,Narosa Publishing
3. William Page, “Using Oracle 8i – Special Edition”, Que/PHI
4. Ivan Bayross, ”SQL & PL/SQL Using Oracle 8i & 9i with SQLJ”, BPB
5. Desai.B, "An introduction to Database Concepts", Galgotia Publications

Course Title:	Software Engineering	**Course Code:**	MCA-2T3-C
Semester:	2nd	**Paper Type:**	Core
Credits:	04	**Max Marks:**	100
Pre Requisite:	MCA-1T1-C, MCA-1T3-C	**Co-Requisite:**	MCA-2T1-C, MCA-2T2-C

Marks Distribution: (Mid Term:30, End Term:50, Viva:10, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To familiarize students with the fundamentals of software engineering.*
- *To introduce students to the basic concepts and principles of structured software engineering.*
- *To make students understand the software management issues.*
- *To introduce the students to the basic concepts of OOSE.*
- *To study advanced concepts in software engineering like Reverse engineering, Reengineering, etc.*

COURSE CONTENT:

UNIT I

Software Engineering: Definition & Evolution, its Role & Impact in Computer Science. Software Process, Characteristics of a SW Process, CMMI, TSP & PSP, Software Product, Characteristics of a Good Software Product, Software Process Models, Comparative Study & Applications. Basic concepts of Agile Process. Software Requirements Analysis (SRA): Requirements - Types, Steps Involved in SRA. SW Requirements Specification (SRS): Need & Characteristics for an SRS, Components of an SRS, Prototype for a Good SRS. Structured Analysis: DFD'S, Control Flow Diagrams, Data Dictionary, State Transition Diagrams, & Entity - Relationship Diagrams. Case Study: Developing a Complete SRS.

UNIT II

Software Design: Concepts & Principles, Design Considerations & Good Design. Characterization of Effective Modular Design (Functional Independence, Cohesion, Coupling). Design: Architectural Design, Procedural Design, Interface Design, & Data Design. SW Architecture Styles: (Dataflow, Call & Return Architectures, Independent Process Architectures, Virtual Machine Architectures). Concept of Verification & Validation. Goals of SW Testing, Testing Principles.

UNIT III

Approaches to the Design of Test Cases: Black Box & White Box Testing, Techniques used by these Approaches: Basis Path & Loop Testing, Graph Based Testing, Equivalence Partitioning, Cyclomatic Complexity, Documentation of Test Cases, Phases in Testing Activity: Unit, Integration, Validation & System Tests. Software Project Management, Phases of Management, Project Planning & Control, Scheduling, Organization & Team Structures, Project Estimation Techniques – KLOC, FP & COCOMO, Risk Analysis & Management, , "Risk Exposure", Software Quality Assurance, Software Configuration Management.

UNIT IV

Technical Metrics for Software. Object Oriented Software Engineering: Object Oriented Paradigm, Concepts - Classes & Objects, Inheritance, Abstraction, Polymorphism, etc. OOA & OOD. Design Methodology – Dynamic Modeling, Functional Modeling. Advanced Concepts: Software Reuse, Re-engineering, Reverse Engineering, Restructuring, Client/Server Software Engineering, Computer Aided Software Engineering, Advances & Future Scope in Software Engineering.

COURSE OUTCOMES:

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

- *Describe software engineering layered technology and process framework.*
- *Understand theories, models, and techniques that provide a basis for the software development life cycle.*
- *Understand software testing approaches including verification and validation, static analysis, reviews, inspections, and audits.*
- *Understand the role of project management including planning, scheduling, risk management, etc.*
- *Work as an individual and/or in team to develop and deliver quality software.*

Text Books:

1. Pressman, Roger, "Software Engineering- A Practitioners Approach", McGraw Hill

References:

1. Gheezi, Jazayeri Et Al, "Fundamentals Of Software Engineering", PHI

2. Ian Sommerville, "Software Engineering", Pearson Education
3. PankajJalote, "An Integrated Approach To Software Engineering",Narosa
4. Peters & Pedrycz, "Software Engineering an Engineering Approach", Wiley

Course Title:	Operating System	**Course Code:**	MCA-2T4-C
Semester:	2nd	**Paper Type:**	Core
Credits:	04	**Max Marks:**	100
Pre Requisite:	MCA-1T1-C, MCA-1T2-C	**Co-Requisite:**	MCA-2T1-C, MCA-2T3-C

Marks Distribution: (Mid Term:30, End Term:50, Viva:10, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To learn the fundamentals of Operating Systems.*
- *To learn the mechanisms of OS to handle processes and threads and their communication*
- *To learn the mechanisms involved in memory management I/O management and file management in OS*
- *To gain knowledge on Mutual exclusion algorithms, deadlock detection algorithms*
- *To know the components and management aspects of concurrency management*
- *To implement the concepts learnt through case studies of different operating systems.*

COURSE CONTENT:

UNIT I

Overview of an Operating system, Functions of Operating System. Process and thread Management Concepts: Process Scheduling, Scheduling Criteria, Scheduling algorithms, Thread Scheduling. Process Synchronization, Semaphores, Critical Section and Monitors, Inter-process Communication, shared memory model and message passing. Deadlocks: Concept of Deadlock, Deadlock prevention, avoidance, detection and Deadlock recovery", Case Studies: LINUX/Windows OS/Android OS (Any one)

UNIT II

Memory Management: Linking, Loading, Memory Allocation, Design Issues & Problems, Fragmentation, Compaction, Memory Management Unit, Paging, Segmentation, Virtual Memory, Demand Paging. Page Replacement Algorithms. Allocation Algorithms, Thrashing Case Studies: LINUX/Windows OS/Android OS (Any one)

UNIT III

File Management: File Structure, File Protection, File System Implementation, Directory Structure, Free Space Management, Allocation Methods, Efficiency & Protection. Case

Studies: UNIX/LINUX/Windows NT OS Disk Management: Disk Structure, Disk Scheduling Algorithm, Disk Management, Swap Space concept & Management, RAID Structure, Disk Performance issues. Case Studies: LINUX/Windows OS/android OS (Any One)

UNIT IV

Multiprocessor Systems: Types of Multiprocessor Operating Systems, Functions & Requirements, Design & Implementation Issues. Distributed Operating System: Difference between Distributed & Centralized Operating System, Advantages & Disadvantages of Distributed Operating System,
Hardware & Software Concepts, Loosely Coupled Systems, Types of Distributed Operating System.

COURSE OUTCOMES:

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

- *Master understanding of design issues associated with operating systems.*
- *Master various process management concepts including scheduling, synchronization, and deadlocks.*
- *Be familiar with various types of operating systems including UNIX.*

Text Books:

1. Tanenbaum, A.S., "Modern Operating System", PHI
2. Peterson, J.L. Abraham, Silberschatz, "Operating System Concepts", Addison Wesley

References:

1. Dietel H.M. "An Introduction To Operating System", Addison Wesley
2. Karnetkar, "UNIX Shell Programming", BPB
3. W.Stallings, "Operating systems"
4. Dhamdhere, "An Operating System –Design & principles"
5. Madnick E, Donovan J, "Operating Systems", TMH
6. Marko Gergent, "Learning android", O'rielly

Course Title:	Advanced Computer Organization & Architecture	**Course Code:**	MCA-2E1-DCE
Semester:	2nd	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-1T2-C	**Co-Requisite:**	MCA-2T1-C, MCA-2T4-C

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To provide an insight about Classification of parallel computers, advanced processor, pipelining, memory design & architecture and multi-processor architecture.*
- *To Understand concepts of parallel processing and design choices of implementing parallel execution within a single processor (pipeline, VLIW, and superscalar) and multiprocessor systems.*
- *To gain knowledge of the state of the art research topics on advanced computing systems.*

COURSE CONTENT:

UNIT I

Computational Models : Introduction , Interpretation of the concept of a computational model , Relationship between the concepts of computational model , programming language & architecture , Basic Computational models , The Von , Neumann computational model ,Key concepts related to computational models .The concept of computer architecture : Evolution & interpretation of the concept of Computer Architecture at different levels of abstraction. The concept of computer architecture at multilevel hierarchical framework.

UNIT II

Introduction to Parallel Processing: Basic Concepts about program, process, thread, process & treads in languages, concurrent & parallel execution, concurrent & parallel programming languages, Types & levels of Parallelism, Classification of Parallel architectures, Basic Parallel Techniques

UNIT III

Introduction to Instruction level Parallel Processors, Evolution & overview, dependencies, instruction scheduling, preserving sequential consistency, Pipelined Processors, Basic Concepts, Pipelined instruction processing. VLIW, Basic Principles, Superscalar processing, introduction, superscalar instruction issue, shelving, parallel execution, preserving the sequential consistency of instruction execution & exception processing

COURSE OUTCOMES:

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

- *Understand the basic structure of computer.*
- *Familiarize the instructions in central processing unit of a computer.*
- *Understand memory organization in a computer.*
- *Understand input/output mechanisms.*
- *Understand parallel processing in a computer.*

Text Books:

1. Advanced Computer Architecture DEZSO SIMA, TERENCE Mountain, PETER KACSUK, Pearson Education, Fifth Indian reprint 2004.

References:

1. V.C. Hamacher. A.G. Vranesic & S. G. Zaky: “Computer Organization”, Tata McGraw Hill.
2. J.P. Hayes: “Computer Architecture & Organization”, McGraw Hill.
3. Morris Mano: “Computer System Architecture”, Pearson Education, 3/e.

Course Title:	Operational Research	**Course Code:**	MCA-2E2-DCE
Semester:	2nd	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-1T3-C	**Co-Requisite:**	MCA-2T1-C, MCA-2T2-C

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To identify and develop operational research models from the verbal description of the real system.*
- *To understand the mathematical tools that are needed to solve optimization problems.*
- *To define and formulate linear programming problems and appreciate their limitations.*
- *To understand how to solve linear programming problems using appropriate techniques and optimization solvers, interpret the results obtained and translate solutions into directives for action.*
- *To conduct and interpret post-optimal and sensitivity analysis and explain the primal-dual relationship.*
- *To develop mathematical skills to analyze and solve integer programming and network models arising from a wide range of applications.*

COURSE CONTENT:

UNIT I

Linear Programming: L P formulation, Graphical methods for LPP with 2 variables, Simplex Algorithm for LPP, Duality theorem in linear programming & applications. Transportation problem: Formulation, methods of selecting initial feasible solutions, Degeneracy & resolution. Assignment problem: Hungarian Method for solving Assignment Problems, Balanced & Unbalanced problems & resolution.

UNIT II

Network Analysis: Shortest routes, Enumeration& applications. Critical Path Method (CPM) & Programme Evaluation and Review Technique(PERT): Use & design of CPM & PERT. Critical Path calculation, Float & its Types, Crashing in Project Management.
Sequencing problems, Johnsons algorithm for processing m jobs through 2, 3 & '*n*' machines. Max flow problem, Min Cut & max-flow min-cut theorems

UNIT III

Replacement & Sequencing models: Replacement of items that fail & deteriorate. Group & individual replacement.
Game theory: definition & explanation, saddle points, Dominance mixed strategies, games without saddle points, 2*N games.
Dynamic Programming: Characteristics of dynamic programming problem, Bellman's optimality principles, dynamic programming under certainty, shortest route problem, Inventory models: introduction to inventory problems & their analytical structure

COURSE OUTCOMES:

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

- *Formulate and solve linear Programming Problems*
- *Determine the optimum solution to constrained and unconstrained*
- *Apply dynamic programming principle to Linear programming problems.*
- *Determine the integer solutions to Linear Programming Problems*

Text Books:

1. N.D.Vohra, "Quatitative Techniques in management", TMH
2. Hamdy A. Taha, "Operations Research: An Introduction", Pearson

References:

1. Sharma J. K., "Operations Research: Theory & Applications", Macmillan India
2. Gross Donald, "Fundamentals of Queuing Theory", 3rd Ed., John Wiley
3. Mokhtar S. Bazaraa, "Linear Programming & Network Flows", John Wiley
4. Hiller Lieberman, "Introduction to Operations Research", TMH

5. Laudon, "Decsion Support Systems", PHI

Course Title:	Microprocessor & Assembly Language Programming	**Course Code:**	MCA-2E3-DCE
Semester:	2nd	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-1T1-C, MCA-1T2-C	**Co-Requisite:**	MCA-2E1-DCE

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To understand the principles of instruction set architecture and assembly language programming.*
- *To understand basic procedures of how a compiler translates C/C++ code to assembly language and perform simple optimizations.*
- *To understand basic principles of interrupt/exception handling.*
- *To explore in detail a simple hardware CPU implementation that supports a small instruction subset.*

COURSE CONTENT:

UNIT I

Software Model of 8088 / 8086 Microprocessor, Memory Add. Space & Data Organization, Data Types, Segment Registers, Memory Segmentation Dedicated, reserved &general use of Memory, generating an Memory Address, Pin-out diagram of 8086 Microprocessor.

UNIT II

The Microcomputer Organization, Assembly Language Programming Development on PC, Instruction Set, Addressing Modes, 8086 Instruction set, Integer Instructions & Computations, Data Transfer, Arithmetic, Logic Shift, Rotate Instruction, Flag Control, Compare, Control Flow & Jump, Subroutine & Subroutine Handling Instructions, Loop &

Loop Handling, String & String Handling Instructions. Statement Syntax for a source Program, Assembler Directives, Assembling, Linking, Loading & executing a run Module.

UNIT III

Isolated I / O, Memory Mapped I/O, DMA Controller, Programming Communication Interfaces Controller. Interfacing I/O devices to microprocessor, programmable peripheral interface. Interrupt, Mechanism, Types & Priority, Interrupt Vector table, Real Mode. 8086 / 8088 Microprocessors, 8086 / 8088 Microprocessor's Minimum Mode, Maximum Mode Systems, Bus Cycle & Unit States, Memory Control Signals, Read & Write Bus Cycles, Memory Interface Circuits.

COURSE OUTCOMES:

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

- *Describe the architecture of 8086*
- *Develop simple program using 8086*
- *Describe the basic peripheral devices and its applications*
- *Differentiate various microprocessor architectures*

Text Books:

1. The 8088 and 8086 Microprocessors: Programming, Interfacing, Software, Hardware and Applications (Prentice Hall international editions) Walter A. Triebel; Avtar Singh

References:

1. DOUGLAS HALL "Microprocessors & Interfacing" Tata McGrawHill.
2. LIU, GIBSON et al "Microcomputer system The 8086/8088 Family" PHI.
3. PAL CHAUDHURI "Computer Organization & Designing" PHI.
4. MORRIS MANO "Computer System Architecture" Pearson Education.
5. GILMORE "Microprocessors" Wiley/ Tata McGraw Hill.

Course Title:	Modelling & Simulation	**Course Code:**	MCA-2E4-DCE
Semester:	2nd	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-1T1-C, MCA-1T2-C	**Co-Requisite:**	MCA-2T1-C

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To understand different methods for random number generation*
- *To understand various probability distributions*
- *To have a clear understanding of the need for the development process to initiate the real problem.*
- *To have a clear understanding of principle and techniques of simulation methods informed by research direction.*
- *To be able to describe the components of continuous and discrete systems and simulate them*
- *To be able to model any system from different fields*
- *To be able to implement numerical algorithm to meet simple requirements, expressed in English*

COURSE CONTENT:

UNIT I

Concepts of Systems, Models, & Simulation. Distributed Lag Model, Cobweb Models, The process of a simulation Study, Exponential Growth Models, Exponential Decay Models, Type of simulation, Discrete-Event Simulation: Time-Advance Mechanisms, Components &—

Organization of a Discrete-Event Simulation Model. Monte Carlo Method. Simulation of Single-Server Queuing System, Simulation of an Inventory System

UNIT II

Continuous Simulation: Pure-pursuit Problem. Monte Carlo Method. Overview of Random numbers and Pseudo random numbers, Random Number Generators: Random Linear Congruently Generators, General Congruences, Composite Random number generator, Testing Random Number Generators. Generating Random Variates: General Approaches, Continuous & Discrete distributions.

UNIT III

Introduction to GPSS, GPSS block-diagrams, General Description, Simulation of a Manufacturing Shop. SNA, Function, Simulation of a Supermarket, GPSS Model of a Simple Telephone System

COURSE OUTCOMES:

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

- *Determine the properties of different types of physical systems and different types of simulations that are suitable to analyze their behaviors;*
- *Analyze data collected from real world and build input models for simulation studies;*
- *Conduct various simulation studies to investigate the behaviors of complex systems;*
- *Conduct statistical analysis of the simulation outputs; and*
- *Analyze discrete event systems through the competent use of computer simulation methods and mathematical modeling techniques.*

Text Books:

1. Averill Law and Averill M. Law, Simulation Modeling and Analysis with Ex-pertÖt Software, Forth Edition, McGraw-Hill Higher Education, 2007
2. Jerry, Banks. Discrete event system simulation. Pearson Education India, 2005.
3. System Simulation – Geoffrey Gordon, 2nd Edition, PHI
4. System Simulation with Digital computer – Narsingh Deo, PHI

References:

1. Law & Kelton, "Simulation Modeling & Analysis", McGraw Hill
2. Fred Maryanski, "Digital Computer Simulation", CBSPD
3. James A. Pyne, "Introduction to Simulation- Programming Techniques & Methods of
4. Analysis", McGraw Hill

5. Zeigler & Kim, "Theory of Modeling & Simulation", Academic Press
6. Banks et al, "Discrete event Simulation", Pearson Education

Course Title:	Computer Graphics	**Course Code:**	MCA-2E5-DCE
Semester:	2nd	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-1T1-C, MCA-1T3-C	**Co-Requisite:**	MCA-2T1-C

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To explain basic primitives like Point, Line, Circle, Ellipse etc. with their mathematical equations coupled with the geometrical interpretations.*
- *To explain process of displaying continuous primitives on Discrete Display Devices.*
- *To explain few transformations like Translation, Rotation, Scaling and composition of these transformations.*
- *To explain Projections from higher dimensional setup to lower dimensions.*
- *To explain drawing of complex scenes using Splines coupled with understanding of spline specifications and few variants of splines besides familiarizing about Beziers.*
- *To acquaint students about Concepts of displaying structures with no inherent regular geometry using Fractals.*

COURSE CONTENT:

UNIT I

An Introduction Graphics System: Computer Graphics & Its Types, Application of computer graphics, Graphics Systems: Video Display Devices, Raster Scan Systems, Random Scan

Systems, Display Buffer, Concept of Double Buffering & Segmentation of Display Buffer. Use of Lookup tables. "Introduction to Color models (RGB, CMY, and HSV)."

UNIT II

Output Primitives & Attributes of Output Primitives: Output Primitives Points & Lines, Line Drawing Algorithms, Circle Generating Algorithms, Scan-Line Polygon Fill Algorithm, Inside-Outside tests, Boundary-Fill Algorithm, Flood Fill Algorithm, Cell Array, Character Generation, Attributes of Output Primitives: Line attributes, Color & Grayscale Levels, Area fill Attributes, Character Attributes, Bundled Attributes. Anti-aliasing.

Two-dimensional Geometric Transformations: Basic Transformations, Matrix Representations & Homogeneous Coordinates, Composite Transformations, Reflection & Shearing.

UNIT III

Two-Dimension Viewing: The viewing Pipeline, Window to view port coordinate transformation, Clipping Operations, Point Clipping, Line Clipping, Polygon Clipping, Text Clipping, Exterior Clipping.

Three–Dimensional Concepts: Three Dimensional Display Methods, 3D Transformations, Parallel Projection & Perspective Projection

Curves & Surfaces, Splines, Spline specification, Interpolated & Approximated Splines. Bezier Splines, Bezier Curves, Cubic Bezier Curves, Bezier Surfaces. B-Splines curves & surfaces. Fractals - Fractal Generation Procedure.

COURSE OUTCOMES:

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

- *Understand concepts of computer graphics.*
- *Understand 2D and 3D transformation, clipping, splines, objects modeling, colour modeling, lighting, textures, visible surface detection.*
- *Understand algorithms to design and create computer graphics scenes.*

Text Books:

1. Hughes, John F., et al. Computer graphics: principles and practice. Pearson Education, 2014.

References:

1. W.M.Newman & Sproull. "Principles of interactive Computer Graphics" ,TMH
2. Steven Harrington." Computer Graphics a Programming Approach" McGraw Hill.
3. Plastock & Kelley. "Schaums outline of theory & problems of computer Graphics"
4. David F Frogers & J Alan Adams. "Procedural Elements of Computer Graphics" McGraw Hill
5. David F Rogers & J Alan Adams. "Mathematical Elements of Computer Graphics" McGraw Hill

Course Title:	Numerical & Statistical Computing	**Course Code:**	MCA-2E6-DCE
Semester:	2nd	**Paper Type:**	DCE
Credits:	03	**Max Marks::**	75
Pre Requisite:	MCA-1T1-C, MCA-1T2-C	**Co-Requisite:**	MCA-2T1-C, MCA-2E2-DCE

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To explain concept of Polynomials, Equations, Zeros and Roots.*
- *To explain few iterative techniques for computation of roots together with their geometrical interpretation and implementation in C.*
- *To explain solving system of equation using known methods along with their geometrical interpretation and Implementation in c.*
- *To explain concepts of Curve Fitting with specific focus on Lagrange's interpolating Polynomial with a prior explanation of using linear and nonlinear curves for curve fitting.*
- *To explain few statistical concepts measures of central tendency, Dispersion etc.*
- *To explain concept of Hypothesis, Null hypothesis besides giving students knowledge about few tests which include T test, F test etc.*
- *To make students ready for courses like Artificial Intelligence and Machine Learning by giving them prior knowledge about Linear Algebra, Probability and statistics.*

COURSE CONTENT:

UNIT I

Approximations & Errors – Types of Programming Errors, Data Errors, Computer & Arithmetic Errors, Round Off and Truncation Errors. Accuracy and Precision, Measures of Accuracy, Error Propagation. Non-Linear Equations, Types of Methods to find solutions to nonlinear equations, Algorithms to Compute Roots of Equation – Methods of Tabulation or Brute Force Method, Method of Bisection, Secant Method, Newton-Raphson Method, Method for False Position. Derivation of mathematical formulas, geometric interpretation and implementation of these methods.

UNIT II

Linear Equations, Types of Methods to find solutions to linear equations. Algorithms to Solve Linear Algebraic Equations: Gauss Elimination, Gauss Jordan, Gauss Seidel, L.U. Decomposition, Lagrange Interpolated Polynomial, Newton'sMethods of INTERPOLATION – Forward difference, Backward difference. Derivation of mathematical formulasand implementation of these methods.

UNIT III

Differential Equations – Concepts and Terminology, Algorithms to solve Ordinary Differential Equations – Euler Method and Modification. The trapezoidal Rule, Simpson's Rule. 4th order R-K Method.
Derivation of mathematical formulas and implementation of these methods.

COURSE OUTCOMES:

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

- *Understand the iterative methods to find solution of polynomial and transcendental equations.*
- *Discuss methods of interpolation and curve fitting.*
- *Find the solution of linear equations using matrices.*

Text Books:

1. Rajaraman, Vaidyeswaran. Computer oriented numerical methods. PHI Learning Pvt. Ltd., 1993.

References:

1. S.C.ChapraandR.P.Canale: "Numerical methods for Engineering". Tata McGraw Hill.
2. Krishenmurty & Sen : "Numerical Algorithms"
3. V. Rajaraman "Computer oriented numerical methods." Prentice Hall of India.
4. McCalla, Thomas Richard: "Introduction to Numerical Methods & FORTRAN Programming", John Wiley & Sons, Inc.
5. Grewal, B. S.: "Higher Engineering Mathematics", Hindustan Offset Problems Series.
6. "SCHAUM'S Solved Problems Series".
7. Sharma, K. D.:"Programming in Fortran".
8. Jain, M. K., Iyengav, S. R. K., Jain, R. K.: "Numerical Methods for Scientific & Engineering Computation"+, Wiley Eastern Ltd, New Delhi.

Course Title:	Java Programming	**Course Code:**	MCA-2E7-DCE
Semester:	2nd	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-1T1-C, MCA-1T3-C	**Co-Requisite:**	MCA-2T1-C

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To cover OOPS concepts like inheritance, polymorphism, overriding etc.*
- *To introduce Lambda expressions, inner classes and interfaces.*
- *To learn about threads, assertions, logging and exception handling.*
- *To know about collection framework and introduction of spring framework.*

COURSE CONTENT:

UNIT I

Fundamental Programming structures in Java: A simple java program, comments, data type, variables, operators, string, input& output, control flow, big numbers, arrays. Objects & Classes: introduction to Object Oriented programming, using predefined classes, defining your own classes, static fields & methods, method parameters, object construction, packages, the class path, documentation comments.

UNIT II

Inheritance Classes superclasses, subclasses, object: the cosmic superclass, generic array list, object wrapping & autoboxing, method with a variable number of parameters, enumeration class, reflection, design hints for inheritance. Interfaces lambda expressions & inner classes. Interfaces, examples of interfaces, lambda expressions, inner classes, proxies. Exceptions, assertion & logging dealing with errors, catching exception, using assertion, logging.

UNIT III

Generic Programming why generic programming, defining a simple generic class, generic methods, bound for type variables, generic code & virtual machine restriction & limitation, inheritance rules for generic types, wildcard types, reflection & generics. Collections Java collection framework, concrete frameworks, Maps, Views & wrappers, Algorithms, Legacy collections

Concurrency, Introduction to threads, Interrupting threads, thread states, thread properties, synchronization, blocking queues, thread safe collections, callable & futures, executors, synchronizers

COURSE OUTCOMES:

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

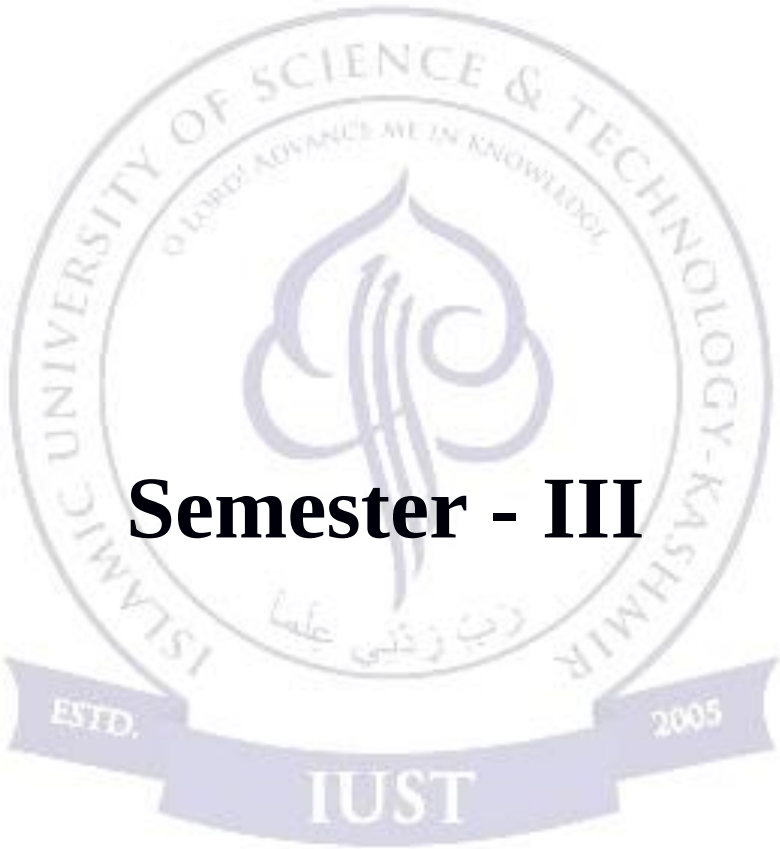
- *Describe the features of Java*
- *Design classes with object-oriented features*
- *Describe advanced features of Java like exception handling, multi-threading etc.*
- *Write programs in JAVA featuring its core capabilities*

Text Books:

1. Core Java Volume 1-Fundamentals by Cay S.horstman 10th edition Publisher: Prentice Hall

References:

1. Java the complete reference by Herbert Schield 10th edition Publisher:Tata Mc Graw Hill
2. Java How to program Dietel and Dietel



Semester - III

Semester-III (26 Credit Semester)						
Course Code	**Course Name**	**Paper category**	**Hours / Week**			**Credits**
			L	T	P	
18 Core Credit Units						
MCA-3T1-C	Analysis and Design of Algorithms	Core	2	1	4	5
MCA-3T2-C	Object Oriented Analysis and Design	Core	2	1	4	5
MCA-3T3-C	Data Communication & Computer Networks	Core	3	1	0	4
MCA-3T4-C	Artificial Intelligence	Core	3	1	0	4
6 Elective Credit Units						
Pool A: 3 Elective Credit Units						
MCA-3E1-DCE	Advanced Operating Systems	DCE	2	1	0	3
MCA-3E2-DCE	Distributed Databases	DCE	2	1	0	3
MCA-3E3-DCE	Bioinformatics	DCE	2	1	0	3
MCA-3E4-DCE	Soft Computing	DCE	2	1	0	3
MCA-3E5-DCE	System Programming	DCE	2	1	0	3
MCA-3E6-DCE	Natural Language Processing	DCE	2	1	0	3
Pool B: 3 Elective Credit Units						
MCA-3E7-DCE	Digital Image Processing	DCE	2	0	2	3
MCA-3E8-DCE	Linux System Programming	DCE	2	0	2	3
MCA-3E9-DCE	Data Warehousing	DCE	2	0	2	3
MCA-3E10-DCE	Open Source Technologies	DCE	2	0	2	3
MCA-3E11-DCE	Introduction to Python	DCE	2	0	2	3
MCA-3E12-DCE	Dot Net	DCE	2	0	2	3
2 credit units to be taken from outside departments						

Course Title:	Analysis & Design of Algorithms	**Course Code:**	MCA-3T1-C
Semester:	3rd	**Paper Type:**	Core
Credits:	05	**Max Marks:**	125
Pre Requisite:	MCA-1T1-C, MCA-2T1-C	**Co-Requisite:**	MCA-3T2-C, MCA-3T4-C

Marks Distribution: (Mid Term:30, End Term:50, Lab:25, Viva:10, Assignment / Presentations:10)

COURSE OBJECTIVE:

The objective of the course is to teach techniques for effective problem solving in computing. The use of different paradigms of problem solving will be used to illustrate clever and efficient ways to solve a given problem. In each case, emphasis will be on rigorously proving correctness of the algorithm. In addition, the analysis of the algorithm will be used to show the efficiency of the algorithm over the naive techniques.

COURSE CONTENT:

UNIT I

Algorithms: Introduction, Importance, Performance Analysis, Designing Algorithms, Growth of Functions, Asymptotic Notations, Solving Recurrences. Review of Data Structures: Stacks, Queues, Trees, Graphs & Hashing.

UNIT II

Divide & Conquer: General Method, Binary Search, MinMax Problem, Merge Sort, Quick Sort, Strassen's Matrix Multiplication. Greedy Method: General Method, Knapsack Problem, Job Sequencing with Deadlines, Minimum Cost Spanning Trees, Optimal Storage on Tapes, Optimal Merge Pattern, Single Source Shortest Paths.

UNIT III

Dynamic Programming: General Method, Multistage Graphs, All Pair Shortest Paths, Traveling Salesperson Problem. Backtracking: General Method, N-Queen Problem, Sum of Subsets Problem, Graph Coloring, 0-1 Knapsack Problem.

UNIT IV

Branch & Bound: General Method, Least Cost Branch & Bound, 8-Queen Problem. Lower Bound Theory: Comparison Trees, Lower Bounds through Reductions, P & NP Problems, NP

Hard & NP Complete Problems, Handling of NP Hard Problems. Parallel Algorithms: Basic Concept, Architectures, Effect of Parallelism.

COURSE OUTCOMES:

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

- *Design and analyze programming problem statements.*
- *Understand the necessary mathematical abstraction to solve problems.*
- *Come up with analysis of efficiency and proofs of correctness.*
- *Comprehend and select algorithm design approaches in a problem specific manner.*

Text Books:

1. Introduction to Algorithms, by Cormen, Leiserson, Rivest, and Stein, MIT Press, Third Edition, 2009.

References:

1. Algorithms, by Dasgupta, Papadimitrou and Vazirani, McGraw-Hill Education, 2006.
2. Computer Algorithms, by Horowitz, Sahni, and Rajasekaran, Silicon Press, 2007.
3. Algorithm Design, by Kleinberg and Tardos, Pearson, 2005.
4. Algorithm Design, by Goodrich and Tamassia, Wiley, 2001.

Course Title:	Object Oriented Analysis and Design	**Course Code:**	MCA-3T2-C
Semester:	3rd	**Paper Type:**	Core
Credits:	05	**Max Marks:**	125
Pre Requisite:	MCA-1T1-C, MCA-2T1-C	**Co-Requisite:**	MCA-3T1-C, MCA-3T4-C

Marks Distribution: (Mid Term:30, End Term:50, Lab:25, Viva:10, Assignment / Presentations:10)

COURSE OBJECTIVE:

- *To understand the definition of problems, requirements specification and the relationship between the specification and the real-world*
- *To Understand systems specification, modelling and a superior functional design and a variety of systems development strategies*
- *To understand the process of transforming Use Cases into Object Oriented software Realizations through OO Analysis and OO Design.*
- *To Understand the essential and fundamental aspects of object-oriented analysis and design, in terms of "how to use" it for the Purpose of specifying and developing software.*

COURSE CONTENT:

UNIT I

Introduction to OOAD, Structures Design & Unified Process, Class & objects, effect of inheritance on polymorphism & variable declarations, concepts that define object orientation.

UNIT II

Requirements: Developing requirements, reviewing requirements, managing requirements, Difficulties & risks in domain & requirement analysis, requirement documents
UML diagrams, Use Case Diagrams, System sequence diagrams, Relationship between sequence diagrams and use cases, Class Diagrams, Interaction Diagrams, State Diagrams, Difficulties & risks in creating class diagram.

UNIT III

Activity Diagrams, Package, component and Deployment Diagrams: Case study – the Next Gen POS system. Architecting & Designing Software: The process of design, design

principles, User interface design. GRASP: Designing objects with responsibilities, Creator, Information expert, Low Coupling, High Cohesion, Controller, Design patterns – Singleton, observer, adapter, Façade, Proxy with examples.

UNIT IV

Mapping design to code: Testing: Issues in OO Testing, Class Testing, OO Integration Testing, GUI Testing, & OO System Testing.

COURSE OUTCOMES:

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

- *Demonstrate the importance of modelling in the software development life cycle.*
- *Become familiar with the Unified modelling Language.*
- *Understand the object-oriented approach to analysing and designing systems and software solutions. Employ the Unified modelling Language notations to create effective and efficient system designs.*
- *Understand the difference between writing programs for the software and doing analysis and design.*
- *Problem formulation and decomposition (analysis) and solution building (design) will be covered.*

Text Books:

1. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development, Third Edition By Craig Larman
2. Analysis Patterns Reusable Object Models, Martin Fowler
3. Design Patterns: Elements of Reusable Object-Oriented Software Book by Erich Gamma, John Vlissides, Ralph Johnson, and Richard Helm

References:

1. Object Oriented Analysis & Design With Applications (Grady Booch ,Robert A.Maksimchuk)
2. Head First-Objected Oriented Analysis & Design
3. Object-Oriented Analysis & Design Using UML (D Jeya Mala, Mc Graw Hill Education)

Course Title:	Data Communication & Computer Networks	**Course Code:**	MCA-3T3-C
Semester:	3rd	**Paper Type:**	Core
Credits:	04	**Max Marks:**	100
Pre Requisite:	MCA-1B1-BC, MCA-1T2-C	**Co-Requisite:**	MCA-3T2-C

Marks Distribution: (Mid Term:30, End Term:50, Viva:10, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To understand networking concepts and basic communication model.*
- *To understand network architectures and components required for data communication.*
- *To analyze the function and design strategy of physical, data link, network layer and transport layer*
- *To acquire basic knowledge of various application protocol for internet security issues and services.*

COURSE CONTENT

UNIT I

Principles of Data Communication, Data Communication Model & Tasks, Concept of Bandwidth & Channel Capacity, Nyquist's Law& Shannon's Law for Quantification of Capacity, Data Rate Versus Baud Rate, Sampling & its types, Nyquist Criterion for Sampling, Data Transmission Concepts, Characteristics of Signals: Digital Signals, Analog Signals (Amplitude, Frequency, Period, Wavelength, Signal to Noise Ratio).

UNIT II

Data Encoding: NRZ-L, NRZ-I Encoding, Multilevel Binary &Biphase Encoding Techniques & their implementations. ASK, FSK, PSK & QPSK. PCM: Sampling & Quantization & Modulation, Delta Modulation, Amplitude Modulation. Reliable Transmission of Data: Asynchronous & Synchronous Transmission, Multiplexing: Concepts & Types (FDM, Synchronous & Statistical TDM). Error Detection: Parity Based, CRC-Based, Checksum Based. Error Control & Recovery Techniques, Concept of ARQ Standard & its Versions.

UNIT III

Goals & Applications of Computer Network. OSI model. TCP/IP protocol suite. LAN: Ethernet & Token Ring. WAN: Concept, Characteristics, & Architecture: WAN subnet. Internetworking Concept & Architectural Model, Connection Oriented & Connectionless Approaches, Packet switching & Circuit switching. Concept of Autonomous Systems. Internet Layer Protocols: IP (Addressing: Classful& classless IP Addressing, IP Multicasting, Routing, Fragmentation & Reassembly).

UNIT IV

ICMP, ARP, RARP. Routing Protocols: Interior (OSPF), Exterior (BGP). Transport Layer Protocols: TCP & UDP. Socket API for Network Programming: Concept of Port & Sockets. Basic Server Architectures. TCP Server & Client. UDP Server & Client. Network Security: Principles of Security, Firewalls & their Components, Encryption Techniques & examples of Encryption Standard (DES, AES & RSA).

COURSE OUTCOMES:

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

- *Describe how to connect machines in a network.*
- *Describe data communication between machines at various locations.*
- *Familiarize with LAN connecting devices and Network Layer.*
- *Learn different Network Layer Protocols.*
- *Understand the concept of Domain Name System (DNS).*

Text Books:

1. Computer Networks ,A.S. Tanenbaum, Pearson Education
2. Data Communications & Networking ,Forouzan, Tata McGraw Hill Company

References:

1. Computer Network, S.S.Shinde, New Age International Publisher.
2. Data & Computer Communication , Shashi Banzal ,Firewall media
3. Data & Computer communication, William Stallings, Pearson

Course Title:	Artificial Intelligence	**Course Code:**	MCA-3T4-C
Semester:	3rd	**Paper Type:**	Core
Credits:	04	**Max Marks:**	100
Pre Requisite:	MCA-1T3-C, MCA-2E7-DCE	**Co-Requisite:**	MCA-3T1-C, MCA-3T2-C

Marks Distribution: (Mid Term:30, End Term:50, Viva:10, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To present an overview of artificial intelligence (AI) principles and approaches.*
- *To develop a basic understanding of the building blocks of AI as presented in terms of intelligent agents: Search, Knowledge representation, inference, logic, and learning.*
- *To have understating of different search problems and their solution using various algorithms.*
- *To have basic understanding of machine learning strategies and their role in Artificial Neural Networks*

COURSE CONTENT:

UNIT I

Introduction to Artificial Intelligence, Applications of Artificial Intelligence, and Intelligent agents: Agents and Environments, the nature of environments, structure of agents, Concept of Rationality. Introduction to First order logic, rules in FOL, Prepositional Logic.

UNIT II

Searching for solutions, uniformed search strategies. Search with partial information (Heuristic search) Greedy best first search, A* search, Memory bounded heuristic search, Heuristic functions.

Local search Algorithms: Hill climbing, local beam search.

Game Playing: Adversial search, Games, minimax, algorithm, optimal decisions in multiplayer games, Alpha-Beta pruning

UNIT III

Artificial Neural networks: Introduction to Artificial neural networks, analogy with biological neural network, McCullough-Pitts Model of a neuron, Single layer and Multilayer perceptron, sigmoid function, Training by back propagation, generalization, avoiding, over fitting. Introduction to Learning, concept of supervised, unsupervised and reinforcement learning.

UNIT IV

Knowledge Representation & Reasoning: Propositional logic, Theory of first order logic, Inference in first order logic, Resolution, Unification, Forward & Backward chaining, Probabilistic reasoning, Utility theory, Hidden Markov Models (HMM), Bayesian Networks.

COURSE OUTCOMES:

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

- *Design intelligent machines/systems which act rationally and take the right decision at the right time.*
- *Define an AI problem and find a solution for it.*
- *Represent Knowledge using various knowledge representation schemes.*
- *Understand the basic knowledge acquisition methods.*
- *Understand Artificial Neural Networks and its applications*

Text Books:

1. Artificial Intelligence – A Modern Approach. Second Edition, Stuart Russel, Peter Norvig, PHI/Pearson Education.

References:

1. Introduction to Artificial Intelligence – Rajendra Akerkar, PHI.
2. Artificial Intelligence, 3rd Edition, Patrick Henry Winston., Pearson Edition
3. Artificial Intelligence and Expert Systems – Patterson PHI
4. Expert Systems: Principles and Programming- Fourth Edn, Giarrantana/ Riley, Thomson

Course Title:	Advanced Operating System	**Course Code:**	MCA-3E1-DCE
Semester:	3rd	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-2T4-C	**Co-Requisite:**	MCA-3T2-C

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To understand the basics of distributed systems.*
- *To understand the distributed communication using RPC*
- *To understand the various issue and challenges in during communication of distributed systems*

COURSE CONTENT:

UNIT I

Introduction to Distributed Systems: Motivation & Goals, Advantages & Disadvantages of Distributed Systems over Centralized Systems, Hardware & Software Concepts. Design Issues: Transparency, Flexibility, Performance, Scalability, Heterogeneity, Reliability and Security. Concept of Client-server Model

UNIT II

Remote Procedure Calling (RPC): Motivation, Mechanism, Complexity, Design Issues & Classification. Synchronization in Distributed Systems: Physical Clock Synchronization (Centralized & Distributed Algorithms), Logical Clocks. Mutual Exclusion: Centralized & Distributed (Contention & Token) Algorithms. Election Algorithms: Bully Algorithm, Invitation Algorithm & Ring Algorithm.

UNIT III

Deadlock Detection: Centralized & Distributed Algorithms. Threads: Characteristics, Advantages & Disadvantages, Design Issues & Usage. System Models: Workstation Model, Processor Pool Model & Hybrid Model. Processor Allocation: Goals & Design Issues. File systems: Hierarchical file systems, Distributed File system (DFS): goals, Requirements, Components, Service Types. SUN-NFS: Design & Issues.

COURSE OUTCOMES:

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

- *Provide hardware and software issues in modern distributed systems.*
- *Get knowledge in distributed architecture, naming, synchronization, consistency and replication, fault tolerance, security, and distributed file systems.*
- *Analyze the current popular distributed systems such as peer-to-peer (P2P) systems*
- *Know about Shared Memory Techniques.*
- *Design and Implement Distributed applications using Technologies like RPC, threads.*
- *Learn how to store data in Distributed File System.*
- *Understand How Distributed Shared Memory is managed.*

Text Books:

1. A. S. Tanenbaum, "Distributed Operating systems", Pearson Education, 2008.

References:

1. Couloris et. al., "Distributed Systems: Concepts and Design", Pearson Education, 2005.
2. P. K. Sinha, "Distributed Operating Systems: Concepts & Design", PHI Learning, 2007.

Course Title:	Distributed Databases	**Course Code:**	MCA-3E2-DCE
Semester:	3rd	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-2T2-C	**Co-Requisite:**	MCA-3T1-C, MCA-3T2-C

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To enhance the previous knowledge of database systems by deepening the understanding of the theoretical and practical aspects of the database technologies and showing the need for distributed database technology to tackle deficiencies of the centralized database systems.*
- *To expose active and emerging research issues in distributed database systems and application development.*

COURSE CONTENT:

UNIT I

Features of Distributed versus Centralized Databases – Why Distributed Databases – Distributed Database Management Systems (DDBMSs)- Review of Databases – Review of Computer Networks-Levels of Distribution Transparency- Reference Architecture for Distributed Databases – Types of Data Fragmentation – Distribution Transparency for read-only Applications – Distribution transparency for Update Applications – Distributed Database Access Primitives – Integrity Constraints in Distributed Databases - A Framework for Distributed Database Design – The Design of Database Fragmentation – The Allocation of Fragments.

UNIT II

Foundations of Distributed Concurrency Control – Distributed Deadlocks – Concurrency Control Based on Timestamps – Optimistic Methods for Distributed Concurrency Control - Reliability – Basic Concepts Non-blocking Commitment Protocols – Reliability and Concurrency Control – Determining a Consistent View of the Network – Detection and Resolution of Inconsistency – Checkpoints and Cold Restart - Distributed Database Administration – Catalog Management in Distributed Databases – Authorization and Protection.

UNIT III

Distributed object database management systems – Fundamental object concepts and Models – Object – Abstract Data Types – Composition (Aggregation) – Class – Collection – Subtyping and Inheritance. – Object Distribution Design – Horizontal Class Partitioning – Vertical Class Partitioning – Path Partitioning – Class Partitioning Algorithms – Allocation – Replication – Alternative Client / Server Architectures – Cache Consistency – Object Identifier Management – Pointer Switching Object Migration – Distributed Object Storage – Object Query Processor Architectures – Query Processing Issues – Query Execution – Correctness Criteria – Transaction Models and Object Structures – Transactions Management in Object DBMSs – Transactions as Objects – Conclusion – Bibliographic Notes – Exercises.

COURSE OUTCOMES:

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

- *Explain the techniques used for data fragmentation, replication, and allocation during the distributed database design process.*
- *Evaluate simple strategies for executing a distributed query to select the strategy that minimizes the amount of data transfer.*
- *Describe distributed concurrency control based on the distinguished copy techniques and the voting methods*
- *Apply theory to practice by building and delivering a distributed database query engine, subject to remote Web service calls.*

Text Books:

1. M.Tamer Ozsu, Patrick Valduriez, Distributed database systems, 2nd Edition, Prentice Hall of India, New Delhi.
2. Distributed Databases:Principles and Systems by Stefano Ceri, McGrawHill

References:

1. Distributed Database Systems by Chhanda Ray, Pearson
2. Distributed Database Management Systems: A Practical Approach by by Frank S. Haug and Saeed K. Rahimi, Wiley
3. Distributed Database Systems by David A. Bell, Jane B. Grimson, and Jane Grimson, Addison-Wesley

Course Title:	Bioinformatics	**Course Code:**	MCA-3E3-DCE
Semester:	3rd	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-1E3-DCE, MCA-2E7-DCE	**Co-Requisite:**	MCA-3T4-C

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To get introduced to the basic concepts of Bioinformatics and its significance in Biological data analysis.*
- *Describe the history, scope and importance of Bioinformatics and role of internet in Bioinformatics.*
- *Explain about the methods to characterize and manage the different types of Biological data.*
- *Classify different types of Biological Databases.*
- *Introduction to the basics of sequence alignment and analysis.*
- *Overview about biological macromolecular structures and structure prediction methods.*

COURSE CONTENT:

UNIT I

Introduction to biology, Living cell: characteristics & functions, basics of molecular biology: DNA, RNA, gene, genetic code, processes of transcription, translation, splicing etc. , basics of proteins, sequences, structure & functions of proteins, Introduction to bioinformatics: Definitions & concepts, & the role of bioinformatics, Types of biological data.

UNIT II

Introduction to databases, types of Biological Databases – flat file databases, relational databases, object-oriented databases, introduction to biological databases**,** Sequence databases (EMBL, GenBank, DDBJ, SWISS-PROT, PIR, TrEMBL), search engines: SRS, ENTREZ. BLAST, FASTA.

UNIT III

Basic overview of algorithms: Space & time complexity. Need & significance of sequence alignment. Global & local alignments, techniques of sequence alignments: Doplot, Needleman-wunsch algorithm for global alignment, Smith-waterman algorithm for local alignments, substitution matrices: PAM, Blosum etc. Gene prediction (ab initio & similarity based). Ontologies in bioinformatics: need for ontologies.

COURSE OUTCOMES:

At the end of the course student will be able to apply:

- *knowledge and awareness of the basic principles and concepts of biology, computer science and mathematics*
- *Existing software effectively to extract information from large databases and to use this information in computer modeling*
- *Problem-solving skills, including the ability to develop new algorithms and analysis methods*
- *An understanding of the intersection of life and information sciences, the core of shared concepts, language and skills the ability to speak the language of structure-function relationships, information theory, gene expression, and database queries*

Text Books:

1. T. K. Attwood & D J Parry-Smith, "Introduction to bioinformatics", Pearson Education
2. Jean-Michel Claveriw, CerdricNotredame, "Bioinformatics – A beginner's Guide", WILEY DreamTech India Pvt

References:

1. Harshawardhan P. Bal." Bioinformatics Principles & Applications" O'Reily
2. T. K. Attwood & D J Parry-Smith, "Introduction to bioinformatics", Pearson Education
3. Jean-Michel Claveriw, CerdricNotredame, "Bioinformatics – A beginner's Guide", WILEY DreamTech India Pvt

4. S.C. Rastogi, N. Mendiratta" Bioinformatics Methods &Applications"TMH
5. Krane ,"Bioinformatic", Pearson Education
6. B. Bergeron, "Bioinformatics Computing", Pearson
7. D. E. Krene& M.L. Payma, "Fundamental concepts of Bioinformatics", Pearson

Course Title:	Soft Computing	**Course Code:**	MCA-3E4-DCE
Semester:	3rd	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-2E7-DCE	**Co-Requisite:**	MCA-3T4-C

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To introduce soft computing concepts and techniques and foster their abilities in designing appropriate technique for a given scenario.*
- *To implement soft computing-based solutions for real-world problems.*
- *To give students knowledge of non-traditional technologies and fundamentals of artificial neural networks, fuzzy sets, fuzzy logic, genetic algorithms.*

COURSE CONTENT:

UNIT I

Soft Computing & AI: Introduction & applications of soft computing, soft computing vs. hard computing, Artificial Intelligence: Introduction & application, Intelligent Agents, Structure of Intelligent Agents & types of agents.

UNIT II

Neural Networks & Fuzzy Logic: Neural networks: Characteristics, artificial neural net terminology, Model of a neuron, Topology, Learning, types of learning, Fuzzy set theory, Fuzzy set versus crisp set, Crisp relation & fuzzy relations, Fuzzy systems: crisp logic, fuzzy logic, introduction & features of membership functions.

UNIT III

Support vector machines, Introduction to Evolutionary computation: Evolutionary algorithms- Genetic & differential evolution, Swarm intelligence, Population based meta heuristic methods,

COURSE OUTCOMES:

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

- *Identify and describe soft computing techniques and their roles in building intelligent machines*
- *Apply fuzzy logic and reasoning to handle uncertainty and solve various engineering problems and to apply genetic algorithms to combinatorial optimization problems.*
- *Evaluate and compare solutions by various soft computing approaches for a given problem.*

Text Books:

1. Jyh:Shing Roger Jang, Chuen:Tsai Sun, EijiMizutani, Neuro: Fuzzy and Soft Computing, Prentice: Hall of India, 2003
2. George J. Klir and Bo Yuan, Fuzzy Sets and Fuzzy Logic: Theory and Applications, Prentice Hall, 1995.

References:

1. Artificial Intelligence-A Modern Approach" – by Stuart Russell, Peter Norvig, , Pearson Education

2. Neural Networks, Fuzzy Logic & Genetic Algorithms, Synthesis & applications, S,Rajasekaranand G.A. VijayalakshmiPai, PHI Publication.
3. Introduction to artificial neural systems - by J.M. Zurada.(Jaico Pub)
4. Artificial Intelligence, Rich E & Knight K, TMH, New Delhi.
5. Neural Network fundamental with Graph , Algo.andAppl, by Bose, TMH
6. Neural Network & Fuzzy System, by Kosko PHI Publication
7. Neural Networks & Fuzzy Logic - by Bart Kosko
8. Neural computing theory & practice - by P.D. wasserman (ANZA PUB).
9. Principles of Soft Computing, S.N. Sivanandamand S.N. Deepa, Wiley Publications
10. Luger G.F. and Stubblefield W.A. (2008). Artificial Intelligence: Structures and strategies for Complex Problem Solving. Addison Wesley, 6th edition.

Course Title:	System Software Design	**Course Code:**	MCA-3E5-DCE
Semester:	3rd	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-2T1-C, MCA-2E3-DCE	**Co-Requisite:**	MCA-3T1-C, MCA-3E1-DCE

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVE:

The objective of this course is to teach procedures for the design of software systems and to provide a basis for judgement in the design of system software. The focus is to discuss the design and implementation of the major system software components with the consideration of underlining the pertinent design issues. The design issues are likely to be discussed in the context of modern computer systems that include the different languages and advanced operating systems.

COURSE CONTENT:

UNIT I

Overview, Introduction to Machine Structure, Evolution of the System Programming Components: Assemblers, Loaders, Macros, Linkers, Compilers.

Evolution of Operating System & User View: Functions, Batch Control Language & Facilities. Machine Structure 360-370: Memory, Registers, Data, Instructions & Special Features. Machine Language & Assembly Language.

UNIT II

Assemblers: General Design Procedure: Problem Statement, Data Structures, Format of Databases, Algorithms, Modularity Lookup. Table Processing: Review of Searching & Sorting Techniques. Macros: Macro Instructions, Features, Conditional Macro Instructions, Macro Calls within Macros, Single & Two Pass Algorithm, Implementation of Macro Calls within Macros. Implementation within an Assembler

UNIT III

Loaders: Schemes: Compile & Go Loaders, General Loader Scheme, Absolute Loaders, Subroutine Linkages, Relocating Loaders. Other Loader Schemes & Binders: Linking Loaders, Overlays & Dynamic Binders. Design of an Absolute Loader, Design of a Direct Linking Loader. Programming Languages & Formal Systems: Significance of High Level Languages, Features. Data Types & Data Structures, Storage Allocation & Scope of Names, Accessing Flexibility, Functional Modularity, Use of Formal Systems, Formal Specification, Formal Grammars, Hierarchy of Languages, Backus Normal Form – BNF, Canonic Systems & Formal Systems.

COURSE OUTCOMES:

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

- *Basic Understanding of Machine Structure and its Major System Software Components*
- *Comprehend Assembler Design Issues and Procedures*
- *Understand Macros and Macro Challenges*
- *Understand various Loader Schemes with the consideration of underlining the pertinent design issues*
- *Understand Compilers in the context of different languages and advanced operating systems.*

Text Books:

1. John J. Donovan, “Systems Programming”,Tata McGraw-Hill

References:

1. Barron.D.W. "Assemblers & Loaders", Mc Donald & Javes
2. Ullman.J.D."Fundamentals of Programming System" Addison & Wesley.
3. M.Dhamdhere."System Programming & Operating Systems"

Course Title:	Natural Language Processing	**Course Code:**	MCA-3E6-DCE
Semester:	3rd	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-2E7-DCE	**Co-Requisite:**	MCA-3T4-C

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To understand natural language processing and to learn how to apply basic algorithms in this field.*
- *To get acquainted with the algorithmic description of the main language levels: morphology, syntax, semantics, and pragmatics, as well as the resources of natural language data - corpora.*
- *To conceive basics of knowledge representation, inference, and relations to the artificial intelligence.*

COURSE CONTENT:

UNIT I

The study of Language, challenges of NLP, The Different Levels of Language Analysis, Representations & Understanding, Linguistic background, Grammars & Parsing, Features & Augmented Grammars, Grammars for Natural Language, Towards Efficient Parsing, Ambiguity Resolution. Applications in Information Retrieval, design features of information retrieval system, Language Modelling: Grammer based model & statistical model.

UNIT II

Word Level & Syntactic Ananlysis: Spelling error detection & correction, parts of speech tagging Morphological parsing, Probabilistic parsing .Semantic Interpretation: Semantics & Logical Form, Linking Syntax & Semantics, Strategies for Semantic Interpretation. Pragmatics: Reference Resolution, Syntactic & Semantic Coherence, Text Coherence, An Inference based resolution algorithm.

UNIT III

Natural language Generation: Introduction to language generation, architecture for generation, surface realization, systemic grammar, functional unification grammar, discourse planning Machine Translation: problems in machine translation, Language Similarities & Differences, transfer metaphor, syntactic transformations, lexical transfer, idea of Interlingua, direct translation, using statistical Techniques.

COURSE OUTCOMES:

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

- *The students will get acquainted with natural language processing and learn how to apply basic algorithms in this field.*
- *They will understand the algorithmic description of the main language levels: morphology, syntax, semantics, and pragmatics, as well as the resources of natural language processing.*
- *They will also grasp basics of knowledge representation, inference, and relations to the artificial intelligence.*

Text Books:

1. James Allen, "Natural Language Understanding", Pearson Publication, ISBN: 978-81-317-08958 2nd Edition

References:

1. D. Jurafsky, J. H. Martin, "Speech & Language Processing", Pearson Education, 2002
2. Speech & Language Processing, by Jurafsky, D. & Martin,

3. J.H.Tanveer Siddiqui, US Tiwary, Natural Language Processing & Information Retrieval

Course Title:	Digital Image Processing	**Course Code:**	MCA-3E7-DCE
Semester:	3rd	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-2E7-DCE	**Co-Requisite:**	MCA-3T4-C

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To study image fundamental and mathematical transforms necessary for image processing.*
- *To study fundamental technologies for digital image, compression, analysis, and processing*
- *To Gain understanding of algorithm, analytical tools, and practical implementations of various digital image applications*

COURSE CONTENT:

UNIT I

Introduction & Digital Image Fundamentals: The origins of Digital Image Processing, Examples of Fields that Use Digital Image Processing, Fundamentals Steps in Image Processing, Elements of Digital Image Processing Systems, Image Representation, Image Sampling & Quantization, Some basic relationships like Neighbors, Connectivity, Distance Measures between pixels, Linear & Non Linear Operations. Image Enhancement in the Spatial Domain: Some basic Gray Level transformations, Histogram Processing, Enhancement Using Arithmetic & Logic operations, Basics of Spatial Filters, Smoothening & Sharpening Spatial Filters, Combining Spatial Enhancement Methods.

UNIT II

Image Enhancement in the Frequency Domain: Introduction to Fourier Transform & the frequency Domain, Smoothing & Sharpening Frequency Domain Filters, Homomorphism Filtering. Image Restoration: A model of The Image Degradation / Restoration Process, Noise Models, Restoration in the presence of Noise Only Spatial Filtering, Periodic Noise Reduction by Frequency Domain Filtering, Linear Position-Invariant Dedradations, Estimation of Degradation Function, Inverse filtering, Wiener filtering, Constrained Least Square Filtering, Geometric Mean Filter, Geometric Transformations.

UNIT III

Image Compression: Coding, Interpixel & Psychovisual Redundancy, Image Compression models, Elements of Information Theory, Error free comparison, Lossy compression, Image compression standards. Image Segmentation: Detection of Discontinuities, Edge linking & boundary detection, Thresholding, Region Oriented Segmentation, Motion based segmentation.

COURSE OUTCOMES:

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

- *Understand Basics of Image formation and transformation using sampling and quantization.*
- *Perform and apply compression and coding techniques used for image data.*

Text Books:

1. Rafael C. Gonzalez & Richard E. Woods, “Digital Image Processing”, Pearson Education

References:

1. A.K. Jain, “Fundamental of Digital Image Processing”, PHI
2. Bernd Jahne, “Digital Image Processing”, Springer
3. William K Pratt, “Digital Image Processing: Piks Inside”, John Wiley & Sons

Course Title:	Linux System Programming	**Course Code:**	MCA-3E8-DCE
Semester:	3rd	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-1T1-C, MCA-2T4-C	**Co-Requisite:**	MCA-3T2-C

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVE:

The objective of course Linux System Programming is to help users in development process and thus maximize the programming time and applications use of Linux system and encourages and enable the users to develop their own real-world applications.

COURSE CONTENT:

UNIT I

Introduction Linux, History of Linux, Linux Architecture & features, Linux and Unix. Introduction to Linux file System. File operations, Mounting and Unmounting of file

systems, Different files types (Ext2 & Ext3). Files & Directory Structure. Overview of Linux process management, process states. Linux Editors and Compilers, Overview of YACC and LEX.

UNIT II

Introduction to Shell programming: Command line programming, Shell types (Bourne Shell, Korn Shell, C shell),Shell Script and structure, Running a Shell Script, Introduction to Linux programming, declaring Variables, input and output operations, Using break, continue, exit, return & Special Parameters ($* and $@) Operators, Conditional Constructs IF-Then-FI,IF-THEN-ELIF-FI, Looping Constructs: for-in, while, untill , Built-in Mathematical Functions,

UNIT III

Filters and Shell Command: Pr, Head, tail, cut, paste, sort, uniq, tr, join etc. grep, egrep, fgrep, sed, awk etc. File permission commands, File and directory command. Pipeline & Input/output redirection and commands, Introduction to system calls, creat, open, close, read etc. background and foreground process managing commands. Linux GUI Development: Introduction to Qt as GUI development tool, Managing Controls and forms in Qt.

COURSE OUTCOMES:

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

- *Use basic fundamental utilities which are required again and again on daily basis to work on a modern operating system.*
- *Write useful shell scripts which greatly and effectively enhance the usefulness of computers, from the point of view of programmers and application developers.*
- *Understand basics of various OS related concepts, from programmer's point of view, like files, directories, kernel, inodes, APIs, system calls, processes, signals, etc.*
- *Develop applications where several processes need to communicate with each other to complete a task.*
- *Use different IPC ways in their programs like Message Queues, Semaphores, and Shared Memories.*
- *Write programs which employs advanced concepts like multithreading.*
- *Write useful programs for networking purposes*

Text Books:

1. A practical guide to Linux commands, Editors and Shell Programming by Mark G.Sobell.
2. System Programming n Linux/Unix by K.C Wang publisher:Springer International
3. Understanding the Linux Kernal Publisher O'Relly Media.

References:

1. Linux System Programming Talking Directly to the Kernel & C Library By Robert LovePublisher: O'Reilly Media Final Release Date: September 2007 Pages: 392
2. Advanced Programming in Unix Environment Second Edition W.Richard Steven Stephen A Rago Publisher Addison Wesley

Course Title:	Data Warehouse	**Course Code:**	MCA-3E9-DCE
Semester:	3rd	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-2T2-C	**Co-Requisite:**	MCA-3E2-DCE

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To provide students with an understanding of the fundamentals of data warehouse, its basic components and the role of data warehouse in decision support.*
- *To provide knowledge of the important steps and techniques to be considered during data warehouse development, future trends and usage of data warehouse.*
- *To introduce the concept of dimensional modelling technique for designing a data warehouse.*
- *To provide knowledge of data integration and the extraction, transformation and load (ETL) processes.*

- *To introduce students to the methods and tools for accessing and analyzing the warehouse data.*

COURSE CONTENT:

UNIT I

Overview of decision support systems: Organizational need for strategic information, Failures of past decision-support systems, operational versus decision-support systems, data warehousing – the only viable solution, data warehouse defined. Data warehouse – The building Blocks: Defining Features, data warehouses & data marts, overview of the components, metadata in the data warehouse
Defining the business requirements: Dimensional analysis, information packages – a new concept, requirements gathering methods, requirements definition: scope & content

UNIT II

Principles of dimensional modeling: Objectives, From Requirements to data design, the STAR schema, STAR Schema Keys, Advantages of the STAR Schema
Dimensional Modeling: Updates to the Dimension tables, miscellaneous dimensions, the snowflake schema, aggregate fact tables.

UNIT III

Data Extraction, Transformation and Loading: Objectives, ETL Overview, ETL Requirements and Steps, Data Extraction- Source Identification, Data Extraction Techniques, Evaluation of Techniques, Data Transformation- Major Transformation Types, Data Integration and Consolidation, Transformation for Dimension Attributes, Transformation Implementation, Data Loading, ETL Tool Options, Other Integration Approaches.

COURSE OUTCOMES:

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

- *Understand the concept related to data warehouse architecture*
- *Compare Online Analytical Processing (OLAP) and Online Analytical Transaction Processing(OLTP) tools*
- *Design a data mart or data warehouse for any organization*
- *Asses raw input data and preprocess it to provide suitable input for range of data mining algorithms*
- *Extract association rules and classification model*
- *Identify the similar objects using clustering techniques*

Text Books:

1. Paul Raj Poonia, "Fundamentals of Data Warehousing", John Wiley & Sons

References:

1. Sam Anahony, "Data Warehousing in the real world: A practical guide for building decision support systems", John Wiley
2. Alex Berson,Stephen J. Smith, "Data Warehousing, Data Mining & OLAP", Tata McGraw Hill W. H. 5.
3. Inmon, "Building the operational data store", John Wiley
4. Kamber & Han, "Data Mining Concepts & Techniques", Hartcourt India P. Ltd.

Course Title:	Open Source Technologies	**Course Code:**	MCA-3E10-DCE
Semester:	3rd	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-1E4-DCE, MCA-2E7-DCE	**Co-Requisite:**	MCA-3T2-C

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To learn about Linux Administration.*
- *To learn PHP and its database connectivity.*
- *To learn how to create Websites with PHP MY-SQL.*
- *To learn Content Management System. And Introduction to No-Sql.*

COURSE CONTENT:

UNIT I

Introduction to Linux – Architecture, Installation, Configuration, Package Management, Basic Linux commands - file system commands, vim. Apache Http Server – installation, basic configuration, setting up virtual hosts, installation of modules, etc

UNIT II

Introduction to PHP - variables, data types, basic constructs, loops, functions, classes. Installation of third party php modules. Introduction to MySQL database. PHP-MySQL connectivity. Introduction to HTTP – HTTP methods, headers, cookies, etc. Creating a basic website with PHP & MySQL.

UNIT III

Content management system based on php & mysql e.g. Wordpress & Drupal – installation, configuration. Introduction to NoSQL – installation & configuration of Mongodb. Basic select, insert, update & delete operations, aggregate operations. Backup & restoration.

COURSE OUTCOMES:

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

- *Explore different open source technology like Linux, PHP & MySQL with different packages.*
- *Execute Linux commands for programming.*
- *Execute programs of PHP with MySQL connection*

Text Books:

1. Core Java Volume 1-Fundamentals by Cay S.horstman 10th edition Publisher: Prentice Hall

References:

1. Java the complete reference by Herbert Schield 10th edition Publisher:Tata Mc Graw Hill
2. Java How to program Dietel and Dietel

Course Title:	Introduction to Python	**Course Code:**	MCA-3E11-DCE
Semester:	3rd	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-2T1-C, MCA-2E7-DCE	**Co-Requisite:**	MCA-3T1-C, MCA-3T4-C

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To learn the essential concepts of Python programming*
- *The course aims at equipping participants to be able to use python programming for solving data science problems.*
- *To understand the advantage of using Python libraries for implementing Machine Learning models*

- *To perform data visualization, web scraping, and natural language processing.*

COURSE CONTENT:

UNIT I

Basics of Python: Introduction to Python, Why learn Python, Keywords and identifiers, comments, indentation and statements, Variables and data types in Python, Operators, Standard Input and Output, Introduction to IDE such as Sublime, pycharm, spyder and relevant packages installations such anaconda.
Control structure and function: if-elif-else, while loop, for loop, break and continue, Introduction to function, Types of functions, Function arguments, Lambda functions, File Handling, packages and modules.

UNIT II

Python Data Structures: Lists, Tuples, Dictionary, Sets, strings, Numpy: Numpy operation, Array and its operation, Matrix and associated operations, Linear algebra and related operations using python. Understand the advantage of using Python libraries for implementing Machine Learning models. Types of data sets.

UNIT III

Pandas data frame and data frame related operations on dataset: Reading data files, pandas data frames, exploratory data analysis, Data preparation and preprocessing (Dealing with missing value, cross-validation, classification, performance measure),
Data visualization on dataset using matplotlib and seaborn libraries: Scatter plot, Line plot, Bar plot, Histogram, Box plot, Pair plot.

Introduction to Regression - Linear, Non-linear, Simple and Multiple regression, and their applications, Introduction to Classification technique - KNN, ANN, Decision Trees and SVM

COURSE OUTCOMES:

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

- *Read, write, and execute simple Python programs.*
- *Write simple Python programs for solving problems.*
- *Decompose a Python program into functions, lists etc.*
- *Read and write data from/to files in Python Programs*
- *Underline the use of package*

Text Books:

1. Mastering Python for data science, Samir Madhavan
2. Introduction to Machine Learning with Python, Andreas C. Mueller

References:

1. Machine Learning using Python, U Dinesh Kumar Manaranjan Pradhan
2. McKinney, W. (2012). Python for data analysis: Data wrangling with Pandas, NumPy, and IPython.

Course Title:	Dot Net	**Course Code:**	MCA-3E12-DCE
Semester:	3rd	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-2T2-C, MCA-2E7-DCE	**Co-Requisite:**	MCA-3T2-C, MCA-3E10-DCE

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *Understand the fundamentals of developing software applications using object oriented methodologies*
- *Creating C# Net applications using standard .net controls.*
- *Connecting to different data sources and querying the database.*

- *To understand the basic of ASP. Net web application.*

COURSE CONTENT:

UNIT I

An Overview of .NET Technology: Features of .NET, Understanding .NET Framework, Main Components of .NET Framework, The Common Language Runtime, The Common Language Specification, The Common Type System, .NET Class Libraries, Assemblies, Metadata and Attributes. Introduction to Visual Studio.NET: Exploring Visual Studio.NET Interface.
C# Variables and Data Types, Program Flow Control in C#, Arrays, Using statement, Namespace , Aliases , The Main() Method, Passing Arguments to main() . Console I/O, Using Comments, Operators and Casts, Error and Exception Handling. C# IDE,

UNIT II

Basic Window Controls: Text Box, Label, Check Box, List Box, Checked List Box, Radio Buttons, Buttons, Tree View and List View Controls.
Objects and Type: Classes and Structs, Partial classes, static classes, Function Overloading, Operator Overloading, Inheritance: Types of inheritance, virtual methods, hiding methods, sealed classes and methods, Interfaces, Derived interfaces. Type safety, Type conversions, boxing and unboxing, comparing objects for equality, type casting, Delegates and Events, Strings, Collections, Array Lists.

UNIT III

Introducing Web Application, Components of Web Application, Building a Web Form, HTML Server Controls. ASP.NET, Using Visual Web Developer, Designing a Simple Web Form. Including C# Code in ASP.NET, Hosting the Web Page, IIS Web Server

An Overview of ADO.NET, Design Goals of ADO.NET, ADO.NET Architecture, Objects Used in ADO.NET Model, .NET Framework Data Providers, .NET Framework Data Provider for SQL Server, Provider for Oracle, ADO.NET Data Set Object Model, Data Binding, Types of Data Binding, Generating Data Set, Binding Controls to the Data Set

COURSE OUTCOMES:

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

- *Understand the Microsoft .NET Framework and ASP.NET page structure*
- *Design web application with variety of controls*

- *Access the data using inbuilt data access tools*
- *Use Microsoft ADO.NET to access data in web Application*
- *Configure and deploy Web Application*
- *Develop secured web application*

Text Books:

1. "ASP .Net : Unleashed" , SAMS Publications'
2. Dietel &Dietel , "C# , How to Program",Pearson Education.

References:

1. "ASP .Net for beginner",Wrox Publications
2. Visual C#.Net by John Sharp & John Jagger, PHI, New Delhi.
3. 14 lesson to get you started with c# and dot net, faraz rasheed

Semester - IV

Semester-IV(24 Credit Semester)						
Course Code	**Course Name**	**Paper category**	**Hours / Week**			**Credits**
			L	T	P	
20 Core Credit Units						
MCA-4T1-C	Theory of Formal Languages	Core	2	2	0	4
MCA-4T2-C	Big Data Analytics	Core	2	2	0	4
MCA-4T3-C	Project work	Core	0	10	10	10
6 Elective Credit Units						
Pool A: 3 Elective Credit Units						
MCA-4E1-DCE	Wireless Communications	DCE	2	1	0	3
MCA-4E2-DCE	Cloud and Grid Computing	DCE	2	1	0	3
MCA-4E3-DCE	Information Security and Networks	DCE	2	1	0	3
MCA-4E4-DCE	Compiler Design	DCE	2	1	0	3
Pool B: 3 Elective Credit Units						
MCA-4E5-DCE	Block Chain Technologies	DCE	2	0	2	3
MCA-4E6-DCE	Machine Learning/Data Science	DCE	2	0	2	3
MCA-4E7-DCE	Quantum Computing	DCE	2	0	2	3
MCA-4E8-DCE	Advanced Java	DCE	2	0	2	3

Course Title: Theory of Computation & Formal Languages **Course Code:** MCA-4T1-C
Semester: 4th **Paper Type:** Core
Credits: 04 **Max Marks:** 100
Pre Requisite: MCA-2T1-C, MCA-3E6-DCE **Co-Requisite:** MCA-4E4-DCE
Marks Distribution: (Mid Term:30, End Term:50, Viva:10, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To be able to construct finite state machines and the equivalent regular expressions.*
- *To understand the class of languages described by finite state machines and regular expressions.*
- *To understand and construct pushdown automata and the equivalent context free grammars.*
- *To understand the concept of construct Linear bounded automata and the equivalent context Sensitive grammars.*
- *To understand the concept of Turing Machine and Universal Turing machine*

COURSE CONTENT:

UNIT I

Introduction to Set Theory, Introduction to Automata, Alphabets, String, Languages, Strings, Kleen Star, Kleen Plus, Introduction to Finite automata, Deterministic finite automata, DFA notations (Transition Table, Transition Table) Languages Accepted by DFA, Designing DFA by pattern recognition, Application of DFA, Non-deterministic Finite Automata, Language accepted by NDFA, Conversion of DFA to NDFA, Overview of e-NDFA. DFA vs NDFA vs e-NDFA,

UNIT II

Finite Automata & Regular Expressions: Definition, Basic Regular Expressions, Obtaining regular expression: Using Basics Expressions & State Elimination Method, Application of regular Expressions, Regular Languages: Definition, Properties, and Pumping Lemma for Regular Languages, Decidable and Closure properties of Regular languages. Limitation of Finite Automata, Equivalence & minimization of DFA.

UNIT III

Introduction to Grammar, Chomsky Hierarchy, Generation of Grammar from Finite Automata & Regular Expressions, Derivation Trees, Left most and Right Most Derivation Trees, Ambiguous Grammar, Context Free Grammar and its applications, properties of context free languages.

Introduction to Push down Automata, Transition & Graphical Representation.

UNIT IV

Introduction to Linear bounded Automata & Turing Machine. Turing machine model, transition table Instantaneous definition, Acceptance and Construction of Turing machine. Standard Turing

machine, Transducers, Lexical and Syntax Analyze: Parsing, Top down Parsing (Predictive parsers, LL (K), recursive Descent) and Bottom up Parsing (LR (K)).

COURSE OUTCOMES:

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

- *Describe the mathematical model of machines.*
- *Understand the concept of formal language and corresponding automaton.*
- *Introduces the concept of ambiguity, derivations and parse tree in grammar.*
- *Apply the acquired knowledge of formal languages to the engineering areas like Compiler Design*
- *Apply the acquired knowledge of finite automata theory and to design discrete problems to solve by computers.*

Text Books:

1. Finite Automata and Formal Languages"- A Simple Approach, A.M Padma Read Pearson Education

References:

1. Hopcroft, J., & Ullman, J., "Introduction to Automata Theory, Languages & Computation", Pearson Education
2. Hopcroft J, R. Motwani, & J. Ullman, "Introduction to Automata Theory, Languages & Computation", Pearson Education
3. P. Linz, "Introduction to Formal Languages & Automata", Jones & Barlett PWS Publishing Company
4. Donald Knuth, "The Art of Computer Programming", Prentice Hall

Course Title:	Big Data Analytics	**Course Code:**	MCA-4T2-C
Semester:	4th	**Paper Type:**	Core
Credits:	04	**Max Marks:**	100
Pre Requisite:	MCA-1E4-DCE, MCA-3T2-C	**Co-Requisite:**	MCA-4E8-DCE

Marks Distribution: (Mid Term:30, End Term:50, Viva:10, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To learn big data challenges in different domains including social media*
- *To learn MAP-REDUCE programming model for better scalability and performance*
- *To analyze the capability of No-SQL systems*
- *To apply machine learning algorithms for big data analytics*

COURSE CONTENT:

UNIT I

Introduction to core concepts and technologies: Introduction, Terminology, data science process, data science toolkit, Types of data, Example applications
Data collection and management: Introduction, Sources of data, Data collection and APIs, Exploring and fixing data, Data storage and management, using multiple data sources

UNIT II

Introduction to Big Data: Big Data - Definition, overview of Big Data, Big Data Characteristics – Volume, Velocity, Variety and other Vs, Issues and challenges of Big Data, Stages of analytical evolution, State of the Practice in Analytics, Big Data Architecture – space of Big Data: Transactions, Interactions, Observations; Big data Technological approaches and Potential use cases for Big Data. The Hadoop Ecosystem–Distributed File Systems basics, Advantages of Hadoop, Query languages for Hadoop,Hadoop Distributed File System (HDFS), Overview of HBase, Hive and PIG.
MapReduce Framework , Introduction to Machine Learning .

UNIT III

Data analysis: Introduction, Terminology and concepts, Introduction to statistics, Central tendencies and distributions, Variance, Distribution properties and arithmetic, Basic machine learning algorithms, Linear regression, SVM, Naive Bayes.
NoSQL Databases – Review of traditional Databases, Need for NoSQL Databases, Columnar Databases, Failover and reliability principles, working mechanisms of NoSQL Databases – HBase, Cassandra, Couch DB, Mango DB.

UNIT IV

Applications of Data Science, Technologies for visualisation, , recent trends in various data collection and analysis techniques, various visualization techniques, application development methods in data science using g R, Spreadsheet-like analytics.

COURSE OUTCOMES:

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

- *Preparing for data summarization, query, and analysis.*
- *Applying data modelling techniques to large data sets*
- *Creating applications for Big Data analytics*
- *Building a complete business data analytic solution*

Text Books:

1. Big Data – A Primer, H. Mohanty, P. Bhuyan, D. Chenthati (Eds.), Springer , Studies in Big Data, vol. 11, 2015.
2. Big Data Analytics with Rand Hadoop, Vignesh Prajapati, PACKT Publishing

References:

1. Mining of Massive Datasets, Jure Leskovec, Anand Rajaraman, Jeffrey D. Ullman, Cambridge Universities Press, 2012.
2. Big Data at Work: Dispelling the Myths, Uncovering the Opportunities Book by Thomas H. Davenpor

Semester:	4th	**Paper Type:**	Core
Credits:	10	**Max Marks:**	250
Pre Requisite:	All courses upto semester 3rd	**Co-Requisite:**	MCA-4T2-C, MCA-4E8-DCE

Marks Distribution: (Mid Term: 100, End Term: 150, Viva: 0, Assignment / Presentations: 0)

PROJECT WORK

Course Title:	Wireless Communications	**Course Code:**	MCA-4E1-DCE
Semester:	4th	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-3T3-C	**Co-Requisite:**	MCA-4E3-DCE

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *The objective of this course is to give an introduction to the fundamentals of the wireless communications systems, the wireless network architectures, protocols, and applications.*
- *To provide an overview of Wireless Communication networks area and its applications in communication engineering.*
- *To enable the student to synthesis and analyze wireless and mobile cellular communication systems over a stochastic fading channel*
- *To provide the student with an understanding of advanced multiple access techniques.*
- *To provide the student with an understanding of diversity reception techniques*
- *To give the student an understanding digital cellular systems (cdma2000, W-CDMA, LTE and candidate 5G waveforms).*

COURSE CONTENT:

UNIT I

Wireless Communication Systems: Basic Concepts, Wireless Networks, Wireless vs Wired networks, Classification of Wireless Networks. Wireless Transmission Concepts: Antennas, Types of antennas, Signal propagation, Multipath propagation. Types of Wireless Telephones (Cordless, Fixed Wireless (WLL), Wireless With Limited Mobility (WLL-M), Fully Mobile Wireless Phones). Introduction to various Generations of Mobile Phone Technologies. Wireline/Wireless Portion of Mobile Communication Network.

UNIT II

Electromagnetic spectrum, its use & allocation to well-known bands. Multiple Access Techniques: FDMA, TDMA & CDMA. Basic concepts of Spread Spectrum(SS) technique; Direct Sequence SS versus Frequency Hopping SS. Concept of cells, sectorization, coverage area, frequency reuse, cellular networks, microcells & handoffs. Simplified Implementation of IS-95 CDMA Using Chip Sequences, Concept of (Walsh) Code, Concept Of CDMA (PCS and Cellular) Channel, Forward & Reverse CDMA Channel for a Cell/Sector, Comparison of Cellular/PCS CDMA Networks, Frequencies & Cell-Sizes.

UNIT III

Cellular Mobile Systems: Introduction & Overview of GSM/CDMA. Channel: Types, Forward & Reverse CDMA channel for a cell/sector (Pilot, Sync, Paging, Forward Traffic Channels, Access, Reverse TC). Concept of Voice Coding. Components of Mobile Network Architecture: MS, BTS, BSC, MSC, & their basic Functions. Types of Handoffs in GSM. Use of HLR & VLR in Mobile
Networks. Concept of Mobile IP & WAP. Wireless LAN Technology: Overview of Wifi & the IEEE 802.11. Concept of Bluetooth & IEEE 802.15.

COURSE OUTCOMES:

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

- *Demonstrate advanced knowledge of networking and wireless networking and understand various types of wireless networks, standards, operations and use cases.*
- *Demonstrate knowledge of protocols used in wireless networks and learn simulating wireless networks.*

Text Books:

1. T. Rappaport, “Wireless Communications, Principles andamp; Practice”, Pearson andy Dornan, “The Essential Guide to Wireless Communications”, Pearson
2. Jochen Schiller, “Mobile Communications”, Pearson andrew Tanenbaum, “Computer Networks”, PHI

References:

1. William Stallings, “Wireless Communications & Networks”, Pearson
2. K.Pahlavan, P.Krishnamurthy, “Principles of Wireless Networks”, Pearson Education.

Course Title:	Cloud & Grid Computing	**Course Code:**	MCA-4E2-DCE
Semester:	4th	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-3E10-DCE	**Co-Requisite:**	MCA-4E5-DCE

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVE:

The objective of studying Cloud and Grid Computing is that it enables new social services by connecting users via social networks that are constructed using multiple cloud service and it will run the software industry upside down. Thus educates potential users about the benefits of cloud computing and the best way to make full use of them.

COURSE CONTENT:

UNIT I

Cloud Computing: Introduction to cloud & Cluster computing, cloud computing vs cluster computing, Evolution of cloud computing, principles of cloud computing, cloud Computing architecture, Cloud computing applications. Cloud service models (IaaS, PaaS, SaaS), could Deployment models (Public, Private, hybrid& community models). Challenge and Security Issues

UNIT II

Grid Computing: Introduction to Grid Computing, Characteristics, grid computing Architecture, grid complexity levels and topologies, grid components and grid layers, applications of Grid Computing. Grid security issues: Authorization and Authentication methods, Grid computing vs Cloud Computing Cluster Computing.

UNIT III

Introduction to OGSA, Services, Schema and architecture, Overview of OGSI, Virtualization: Virtual machines and visualization of clusters and data centres, levels of virtualization, virtualization structures and tools and mechanism, virtualization of CPU, Memory & I/O devices

COURSE OUTCOMES:

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

- *Elaborate the basic concepts of cloud computing and defining the basic terms*
- *Understand various cloud implementations and migration techniques*
- *To define the various industrial applications of cloud virtualization.*
- *learn security challenges and preventive measures in cloud computing*

Text Books:

1. Cloud Computing by pankaj Sharma published by S.K.Kataria and Sons.
2. Cloud Computig Bible: Published by Wiley India Pvt. Ltd.
3. Grid and Cloud Computing and Applications: The 2014 World Comp International Conference. Publisher C.S.R.E.A
4. Grid, Cloud and Cluster Computing and Applications by Hamid R.Arabnia, Ashu M.G.Solo

References:

1. Mastering Cloud Computing, Paperback-1 Feb 2013 by Buyya, Vecchiola&Selvi.
2. Cloud Computing: Concepts, Technology & Architecture, 1e Paperback-2014 by Erl.
3. Grid & Cloud Computing, a business perspective on Technology & Applications, Springer by Stanoevska-Slabeva, Katarina, Wozniak, Thomas, Ristol, Santi (Eds.)

Course Title:	Information Security & Networks	**Course Code:**	MCA-4E3-DCE
Semester:	4th	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-3T3-C	**Co-Requisite:**	MCA-4E1-DCE

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To develop an understanding of information assurance as practiced in computer operating systems, distributed systems, networks and representative applications.*
- *To Gain familiarity with prevalent network and distributed system attacks, defenses against them, and forensics to investigate the aftermath.*
- *Develop a basic understanding of cryptography, how it has evolved, and some key encryption techniques used today.*
- *Develop an understanding of security policies (such as authentication, integrity and confidentiality), as well as protocols to implement such policies in the form of message exchanges.*

COURSE CONTENT:

UNIT I

Introduction: Security trends, the OSI Security Architecture, Security attacks, Security mechanisms, Security services, Model for Network Security. Encryption: Symmetric & Asymmetric Encryption, Symmetric Cipher model, Substitution techniques (Mono-alphabetic & Poly-alphabetic) & Transposition Technique, Rotor Machines, Block and Stream ciphers, Modern Round Ciphers: DES, AES, Introduction to Steganography. Encryption vs Steganography.

UNIT II

Introduction to Number Theory: Prime Numbers, Fermat's & Euler's theorem & discrete logarithms. Public Key Cryptography & RSA. Key management: Diffie-Hellman Key Exchange, Elliptic Curve Arithmetic & Elliptic Curve Cryptography. Message Authentication & Hash functions. Digital Signatures & Authentication protocols. Network Security Applications: IP Security

UNIT III

Mechanics of Routing Protocols: Routing , Static and Dynamic Routing, Distant vector routing & link state routing, Internet working with Dissimilar Protocols, Protocol design Consideration: Simplicity, flexibility, Optimality, Overhead & Scaling, Operation above Capacity, forward compatibility. Migration: Routing Algorithms & addressing parameters, making multi-protocol operation possible. Network Security: Security Features in Wireless Adhoc & Wireless Sensor Networks.

COURSE OUTCOMES:

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

- *Evaluate the security treats in modern computer era*
- *Define and identify firewall and network filtering*
- *List and recognize various VPN*
- *Identify different technique of sandboxing*
- *Distinguish various ethical hacking and testing procedures*

Text Books:

1. William Stalling ," Cryptography & Network Security", Pearson Education

References:

1. Radia Perlman, "Interconnections: Bridges, Routers switches & Internet-working protocols Pearson education
2. Mark Sportack, "IP Routing Fundamentals", Pearson Education
3. Gerard J. Holzmann, "Design & Validation Computer Protocols", Prentice Hall

Course Title:	Compiler Design	**Course Code:**	MCA-4E4-DCE
Semester:	4th	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-3E5-DCE	**Co-Requisite:**	MCA-4T1-C

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To understand the various phases in the design of a compiler.*
- *To understand the design of top-down and bottom-up parsers.*
- *To understand syntax directed translation schemes.*
- *To learn to develop algorithms to generate code for a target machine.*

COURSE CONTENT:

UNIT I

Introduction to compilers: Structure of a compiler – Lexical Analysis – Role of Lexical Analyzer – Input Buffering –Specification of Tokens – Recognition of Tokens – Lex – Finite Automata – Regular Expressions to Automata – Minimizing DFA.

UNIT II

Syntax Analysis: Role of Parser – Grammars – Error Handling – Context-free grammars – Writing a grammar –Top Down Parsing – General Strategies Recursive Descent Parser Predictive Parser-LL(1) Parser-Shift Reduce Parser-LR Parser-LR(0)Item Construction of SLR Parsing Table -Introduction to LALR Parser – Error Handling and Recovery in Syntax Analyzer
Intermediate Code Generation: Syntax Directed Definitions, Evaluation Orders for Syntax Directed Definitions, Intermediate Languages: Syntax Tree, Three Address Code, Types and Declarations, Translation of Expressions, Type Checking.

UNIT III

Run-Time Environment And Code Generation: Storage Organization, Stack Allocation Space, Access to Non-local Data on the Stack, Heap Management-– Issues in Code Generation – Design of a simple Code Generator.
Code Optimization: Principal Sources of Optimization – Peep-hole optimization – DAG - Optimization of Basic Blocks Global Data Flow Analysis – Efficient Data Flow Algorithm.

COURSE OUTCOMES:

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

- *Get an understanding of how the different phases of compiler work.*
- *Study of lexical analysis, and various parsing techiques.*
- *Use compiler tools like lex and yacc.*
- *Implement a simple compiler for a hypothetical machine using syntax directed translation*

Text Books:

1. Compilers: Principles, Techniques and Tools, Second Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffry D. Ullman, Pearson.
2. Compiler Construction-Principles and Practice, Kenneth C Louden, Cengage Learning.

References:

1. Modern compiler implementation in C, Andrew W Appel, Revised edition, Cambridge University Press.
2. The Theory and Practice of Compiler writing, J. P. Tremblay and P. G. Sorenson, TMH

Course Title:	Block Chain Technologies	**Course Code:**	MCA-4E5-DCE
Semester:	4th	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-3E2-DCE, MCA-3E10-DCE	**Co-Requisite:**	MCA-4E2-DCE

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To understand the function of Block chain as a method of securing distributed ledgers*
- *To familiarize the functional/operational aspects of cryptocurrency ecosystem.*
- *To familiarize about wallets and learn their utilization of wallet during transaction*
- *To understand that how to write and apply the Smart Contracts*
- *To understand the concept of Hyperledger*

COURSE CONTENT:

UNIT I

Introduction to Cryptography, Introduction to graph, ring and field, prime and relative prime numbers, modular arithmetic, Fermat's and Euler's theorem, Euclid's Algorithm, RSA algorithm, Diffie-Hellman key exchange algorithm, ElGamal Encryption, Elliptic curve cryptography, SHA 256, Digital Signature, Zero Knowledge Proof (ZKP)

UNIT II

Introduction from barter system to Cryptocurrency, fundamental of Blockchain, Block structure, Genesis Block, Orphaned Blocks, Stale Block, Uncle Block, Distributed Ledger Technology (DLT), peer-to- peer network, Merkle Tree, Lifecycle of Blockchain, Evolutions of Blockchain, Fork, double spending money, Transactions and UTXO's, Types of Blockchain. Need of Blockchain, Benefits of Blockchain

UNIT III

Cryptocurrencies: BitCoin (BTC), Ethereum (ETH), Ripple (XRP), LiteCoin (LTC), Bitcoin Cash (BCH), Mining pools, Mining, Difficulty Level, Current Target, Nonce, how miners picks transactions, How do mempools work, 51% attack.

Consensus Algorithms: Proof of Work (PoW), Asynchronous Byzantine Agreement, Proof of Stake (PoS), Hybrid models (PoW + PoS),

COURSE OUTCOMES:

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

- *Implement the blockchain*
- *Implement the smart contracts on Ethereum platform.*
- *Implement the use cases on Hyperledger*

Text Books

1. Mastering Blockchain, Imran Bashir, Packt Publishing
2. Bitcoin and Cryptocurrency Technologies, Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, Steven Goldfeder, Princeton University Press. https://bitcoinbook.cs.princeton.edu/

References:

1. Grokking Bitcoin, Kalle Rosenbaum, Manning Publications.
2. http://rosenbaum.se/book/grokking-bitcoin.htmlModern compiler implementation in C
3. Blockchain Basics, Daniel Drescher, Apress Publication

Course Title:	Machine Learning	**Course Code:**	MCA-4E6-DCE
Semester:	4th	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-3T4-C	**Co-Requisite:**	MCA-4T1-C, MCA-4E5-DCE

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To explain Curve fitting with a specific focus on Regression analysis, logistic Regression Analysis.*
- *To explain different machine learning procedures which include Supervised, Unsupervised, semi supervised and reinforcement learning etc.*
- *To explain familiar machine learning Techniques like Support Vector Machines and its nonlinear variant.*
- *To explain few distribution like Gaussian, Dirichlet etc. and its subsequent use in modelling data for learning.*
- *To explain common applications of machine learning with a specific focus on latest developments in the said area.*

COURSE CONTENT:

UNIT I

Introduction: Basic definitions, learning types: supervised & unsupervised, hypothesis space & inductive bias, evaluation, cross-validation, Linear regression, Decision trees, over fitting, Instance based learning, Feature reduction, Collaborative filtering based recommendation

UNIT II

Probability & Bayes learning Logistic Regression, Support Vector Machine, Kernel function & Kernel SVM neural network: Perceptron, multilayer network, back propagation, introduction to deep neural network, Introduction to Graphical Models. Generative Vs. Discriminative Models.

UNIT III

Computational learning theory, PAC learning model, Sample complexity, VC Dimension, Ensemble learning, Clustering: k-means, adaptive hierarchical clustering, Gaussian mixture model. Some application areas of machine learning e.g. Natural Language Processing, Computer Vision, applications on the web.

COURSE OUTCOMES:

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

- *Apply a variety of learning algorithms to data.*
- *Perform evaluation of learning algorithms and model selection?*
- *Apply machine learning algorithms to solve problems of moderate complexity*

Text Books:

1. Michalski, Ryszard S., Jaime G. Carbonell, and Tom M. Mitchell, eds. Machine learning: An artificial intelligence approach. Springer Science & Business Media, 2013.

References:

1. Machine Learning. Tom Mitchell. First Edition, McGraw- Hill, 1997.
2. Introduction to Machine Learning Edition 2, by Ethem Alpaydin

Course Title:	Quantum Computing	**Course Code:**	MCA-4E7-DCE
Semester:	4th	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-3E10-DCE	**Co-Requisite:**	MCA-4E6-DCE

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

The course will provide an insight of basic of quantum physics from a computer scientist's perspective, and how it describes reality and understand the philosophical implications of quantum computing

COURSE CONTENT:

UNIT I

Qubit & Quantum States: The Qubit, Vector Spaces. Linear Combination Of Vectors, Uniqueness of a spanning set, basis & dimensions, inner Products, orthonormality, gram-schmidt orthogonalization, bra-ket formalism, the Cauchy- schwarez and triangle Inequalities.

UNIT II

Matrices & Operators: Observables, The Pauli Operators, Outer Products, The Closure Relation, Representation of operators using matrices, outer products & matrix representation, matrix representation of operators in two dimensional spaces, Pauli Matrix, Hermitian unitary and normal operator, Eigen values & Eigen Vectors, Spectral Decomposition, Trace of an operator, important properties of Trace, Expectation Value of Operator, Projection Operator, Positive Operators.

UNIT III

Commutator Algebra, Heisenberg uncertainty principle, polar decomposition & singular values, Postulates of Quantum Mechanics.

COURSE OUTCOMES:

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

Achieve knowledge of Vector spaces, Matrices, Quantum state, Density operator and Quantum Measurement theory.

Text Books:

1. Quantum Computing without Magic by Zdzislaw Meglicki
2. Quantum Computing Explained By DAVID Mc MAHON

References:

1. Quantum Computer Science By Marco Lanzagorta, Jeffrey Uhlmann
2. An Introduction to Quantum Computing Phillip Kaye, Raymond Laflamme, Michele Mosca.

Course Title:	Advanced Java	**Course Code:**	MCA-4E8-DCE
Semester:	4th	**Paper Type:**	DCE
Credits:	03	**Max Marks:**	75
Pre Requisite:	MCA-3T2-C	**Co-Requisite:**	MCA-4T2-C

Marks Distribution: (Mid Term:25, End Term:35, Viva:05, Assignment / Presentations:10)

COURSE OBJECTIVES:

- *To learn about streams and how java is using functional programming concepts.*
- *To learn about database connectivity in java.*
- *To learn about socket programming in java.*
- *To learn about Spring Framework in java*

COURSE CONTENT:

UNIT I

Streams: stream operations, stream creation, filter map & flatmap methods, extracting substreams & concatenating streams, other stream transformation, simple reductions, the optional type ,collecting results, collecting into maps, grouping & partitioning, downstream collections, reduction operations, primitive type streams, parallel streamsInput & outputi/o streams, text input & output, reading & writing binary data, object i/o streams & serialization.

UNIT II

Networking Connecting to a server, implementing servers, Database Programming JDBC design, structured query language, JDBC Configuration, Working with JDBC statements, query execution, scrollable & updatable result sets, row sets, metadata, transactions, advanced sql type, connection management in web & enterprise applications

UNIT III

Core Spring: Introduction to spring, wiring beans, advanced wiring, aspect oriented spring, Spring on web building, spring web application , Spring in the backend using databases with Spring & JDBC, persisting database with object relational mapping, advanced spring MVC, working with spring web flow , securing web applications.

COURSE OUTCOMES:

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

- *Interpret enumerations and collections in advanced Java.*
- *Build programs using collection framework.*
- *Illustrate and develop String Handling methods in JAVA.*
- *Apply Servlets to develop web applications.*
- *Demonstrate database access using JDBC API.*
- *Design reusable software components using JSP.*

Text Books:

1. Core Java Volume 11-Advanced features by Cay S.horstman 10th edition Publisher: Prentice HallSpring in Action, Fourth Edition

References:

1. Pro Spring 5 Authors: Cosmina, I.Harrop, R., Schaefer, C.
2. Spring Boot in Action by Craig Walls